

Automatikus esemény detektálás megvalósítása és fejlesztési lehetőségei

Luspay Tamás (BT6H8T)

2006. május

TARTALOMJEGYZÉK

<i>Bevezetés</i>	1
<i>1. Automatikus Esemény Detektálás - irodalom áttekintése</i>	3
1.1. A teljesítmény mérésére szolgáló paraméterek	3
1.2. Eseményérzékelő algoritmusok	5
1.2.1. Összehasonlító vagy minta felismerő algoritmusok	5
1.2.2. Forgalom modell és Elméleti Algoritmusok	7
1.2.3. Statisztikai illetve idősor alapú algoritmusok	8
1.2.4. Mesterséges Intelligencia használata eseményérzékeléshez	9
1.3. Videokamerák felhasználása eseményérzékelésre	10
<i>2. Az autópálya forgalom dinamikus modellje</i>	11
2.1. Modell egyenletek levezetése	11
2.2. Valós adatok, a vizsgált helyszín	14
2.2.1. Az adatok beolvasása	16
2.3. Modellparaméterek "off-line" kalibrációja	17
2.3.1. Szabad áramlási sebesség, kritikus forgalomsűrűség, modellparaméter meghatározása	17
2.3.2. Modell validáció	19
2.3.3. Modell értékelés	21
<i>3. Lineáris Kalman-szűrő</i>	23
3.1. Egyenletek levezetése	23
3.1.1. A diszkrét lineáris Kalman-szűrő algoritmus	24
3.2. A lineáris Kalman-szűrő hangolása	25
3.2.1. A Kalman-szűrő frekvencia-tartománybeli tulajdonsága	25
3.2.2. Szinguláris értékek	26
3.2.3. Hangolási lehetőségek	27
3.3. Lineáris Kalman-szűrő felhasználása forgalmi változók becslésére	28
3.3.1. A modell linearizálása	28
3.3.2. Modellezés	31
3.3.3. A lineáris Kalman-szűrő hangolása, frekvenciatartománybeli analízis	32
3.3.4. Lineáris Kalman-szűrő baleseti körülmények mellett	35

4.	<i>Kiterjesztett Kalman-szűrő</i>	37
4.1.	Egyenletek levezetése	37
4.1.1.	Kiterjesztett Kalman-szűrő algoritmus	39
4.2.	Kiterjesztett Kalman-szűrő alkalmazása forgalmi változók becslésére	40
4.3.	Paraméterek becslése kiterjesztett Kalman-szűrővel	43
5.	<i>Automatikus Esemény Detektálás a becsült forgalmi változók felhasználásával</i>	46
5.1.	Sebességgradiens módszer	46
5.2.	Esettanulmány	48
5.3.	Értékelés	49
5.4.	Továbbfejlesztési lehetőségek	51
	<i>Összefoglalás</i>	53
	<i>Függelék</i>	57
A.	<i>MATLAB Scriptek</i>	i
A.1.	megnyitas.m	i
A.2.	kritikus.m	i
A.3.	tsim.m	i
A.4.	metanetmodell.m	ii
A.5.	Kalman.m	ii
A.6.	ExKalman.m	iii
A.7.	EstimPara.m	iii
A.8.	aid.m	iii
B.	<i>MATLAB SIMULINK ábrák</i>	iv

Használt rövidítések, jelölések

AID - Automatic Incident Detection

DR - Detection Rate

TTD - Time To Detect

MTTD - Mean Time To Detect

FAR - False Alarm Rate

PI - Performance Index

KFDI - Kalman Frequency Domain Identity

SISO - Single Input Single Output

MIMO - Multiple Input Multiple Output

MHE - Moving Horizon Estimation

CARE - Control Algebraic Ricatti Equation

FARE - Filter Algebraic Ricatti Equation

BEVEZETÉS

Diplomatervem témája egy automatikus eseményérzékelő rendszer felépítése és vizsgálata.

Az autópályán történő balesetek károkat és forgalmi torlódást okoznak, növelik a károsanyagkibocsátást és évente milliókba kerülnek a késések, költségek, járműrongálódások és személyi sérülések. Balesetek esetén a reagálási idő (a baleset bekövetkezése és a mentőcsoporthelyszínre érkezése között eltelt idő) minimalizálása számos szempontból kritikus. A legfontosabb a sérültek kezelése. A gyorsabb kiérkezés növeli a baleset súlyos sérültjeinek túlélési arányát. Másodsorban, a baleseti körülmények gyors felszámolása minimalizálja a forgalmi áramlat megszakítását és a további lehetséges balesetek számát. Automatikus Esemény Detektálás alatt (Automatic Incident Detection - AID) a lehetséges balesetek gyors érzékelésére kidolgozott módszereket értjük. A technológiát az 1970-es évek óta folyamatosan tesztelik, fejlesztik. Ezalatt az idő alatt számos eseményérzékelő algoritmust fejlesztettek ki. Idővel, a technológia folyamatos fejlődésével lehetővé vált videokamerák alkalmazása is, amelyeknél a kezdeti "személyes távfelügyelet" helyett fokozatosan automatikus programok vették át a funkciókat.

Bár külföldön már több évtizedes hagyománya van a hasonló rendszereknek, azonban Magyarországon, sajnálatos módon még szinte ismeretlen területnek számít az AID témaköre. Remélhetőleg jelen diplomaterv - hacsak kis mértékben is - de hozzájárulhat az ezirányú kutatások és esetleges konkrét megvalósítások hazai elterjedésének.

A rendelkezésre álló irodalom áttekintése és csoportosítása után az autópálya forgalom dinamikus modellén alapuló esemény detektálást választottam. A modell összetettsége és makroszkopikus szemléletmódja alapján megfelelő színvonalú eseményérzékelésre alkalmas, így alapjául szolgálhat a felépítendő AID rendszernek. Problémát jelent azonban a rendelkezésre álló adatok térbeli eloszlása, mennyisége. Az autópályákon elhelyezett induktív hurok detektorok illetve egyéb érzékelő berendezések ugyanis csak egy adott ponton szolgáltatnak információt a forgalom lefolyásáról. Bár léteznek rendszerek melyek ezen adatokat felhasználva próbálják meg érzékelni a bekövetkezett eseményeket, azonban a jobb teljesítmény elérése érdekében állapotbecslés alkalmazásával információt kaphatunk a forgalom lefolyásáról a detektorok közötti autópálya szakaszokon is.

Az állapotbecslés megvalósítására Kalman-szűrési technikát alkalmaztam, mellyel a detektorok közötti autópálya szakasz forgalmáról is megfelelő becslés adható, ami elősegítheti a bekövetkezett balesetek illetve más abnormális jelenségek precíz érzékelését. A tervezett Kalman-szűrő által szolgáltatott adatok felhasználásával valósítottam meg a kiírt diplomaterv feladatát. Vagyis a forgalom dinamikus modelljén alapuló állapotbecslési eljárás kidolgozásá-

val kapott információkat használtam fel.

Az AID rendszer, melyet kidolgoztam dinamikus feltételeken alapszik, szemben statikus értékekkel. Ezáltal megkülönböztethetőek a visszatérő torlódások illetve a hirtelen változást jelentő események okozta torlódások. A forgalomlefolysis alapvető viselkedésén alapuló, makroszkopikus eljárást dolgoztam ki, mely eljárásról a szakirodalomban nem olvastam.

Ezúton szeretném megköszönni konzulensemnek Dr. Kulcsár Baláznak a nekem nyújtott számtalan segítséget. Továbbá köszönöm vállalati konzulensemnek Borbély Árpádnak az Állami Autópálya Rt. munkatársának segítőkészségét és a szükséges adatok illetve információk hozzáférhetőségének biztosítását.

1. AUTOMATIKUS ESEMÉNY DETEKTÁLÁS - IRODALOM ÁTTEKINTÉSE

A fejezet témája az AID rendszerekről szóló irodalom rövid áttekintése és összefoglalása, betekintést nyerve azok igen sokszínű kialakításába.

A fejezet első részében az Automatikus Esemény Detektáló rendszerek teljesítményét mérő paramétereiről lesz szó. Az Érzékelési Arány, az Érzékelési Idő és a Téves Riasztások Aránya a legfontosabb jellemzői az AID rendszereknek, algoritmusoknak. A három mutatószám definiálása és értelmezése után világos képet alkothatunk a rendszerekkel szemben támasztott legfőbb követelésekről. Az első rész végén ezen paraméterek közti összefüggésről lesz szó. Majd a szakirodalomban elfogadott Teljesítmény Index által numerikusan is összehasonlíthatjuk az algoritmusokat.

A második rész tartalmát tekintve a legfontosabb algoritmus csoportokat mutatja be. Először a hurokdetektorral működő rendszerekbe tartozó algoritmusok kerülnek bemutatásra. Az egyes csoportokat egy-egy jellemző példával, konkrét működési algoritmusok bemutatásával szemléltetem. Utána a videokamerák használatával működő rendszerekről lesz szó, az egyes csoportok működési elvének bemutatásával.

1.1. A teljesítmény mérésére szolgáló paraméterek

Három paraméter használatos az eseményérzékelő algoritmusok teljesítményének mérésére. Sajnos, ezen paraméterek definiálása az egyes tanulmányokban eltérő. E három paraméter a következő: Érzékelési Arány (DR - Detection Rate), Téves Riasztási Arány (FAR - False Alarm Rate) és az Érzékelési Idő (TTD - Time To Detect).

Érzékelési arány

Az érzékelési arány általánosan az érzékelt események számát viszonyítja az összes bekövetkezett eseményhez képest. Ez az arány az esemény értelmezésétől függően változhat. Néhány tanulmány minden leállt járművet eseményként kezel, az elakadás helyétől függetlenül, míg mások csak a sáv-elzáró eseményeket veszik figyelembe. Ebből következően ez utóbbi értelmezés szerinti értékeléssel nagyobb érzékelési arány érhető el, hiszen az út szélén történő balesetek, leállások legtöbbször nem okoznak a forgalmi áramlatban olyan mértékű változást, hogy a rendszer riasztást indítson.

Érzékelési idő

Az érzékelési időt az esemény bekövetkezésétől annak érzékeléséig eltelt idővel definiálhatjuk. Itt kell megjegyezni, hogy az algoritmusok gyakran figyelembe veszik a riasztások tartósságát. Ez a tartósság fejezi ki azoknak az időintervallumoknak a számát, melyekben a forgalom "esemény-szintű" megszakításának fenn kell állnia mielőtt a rendszer riasztana. Ez az időköz a legtöbb algoritmus esetében 20-30 másodperc. (Ezen kívül használatos egy másik paraméter is, a Mean Time To Detect, vagyis a Közepes Érzékelési Idő, amely n eseményre vonatkozó érzékelési idők számtani átlaga.)

Téves riasztási arány

A téves riasztási arány általánosan a hibás érzékelési jelek százalékos arányaként definiált az algoritmusok által hozott döntésekhez viszonyítva. A legtöbb algoritmus mindegyik időközben egy döntést hoz érzékelő állomásonként. Ezen értékek ellentmondásosak az irodalomban, mivel számításuk több eltérő módon történhet. A fentebb megadott értelmezéssel alacsony téves riasztási arányokat jelentettek. Azonban a rendszer által hozott döntések illetve a hálózatbeli detektor-párok száma sok téves riasztást okozhat rövid időn belül. Más rendszerek viszonylag kevés döntést hoznak időközönként, s ezzel nagyobb Téves Riasztási Arányt produkálnak. Egy másik értelmezési mód az időegységre és érzékelő-állomásra eső téves riasztások száma.

A jelzőszámok közti összefüggés és a Teljesítmény Index

Ezen mérések értékei kölcsönösen egymástól függőek. Általánosan elmondhatjuk, hogy az érzékelési arány növelésével a téves riasztások aránya is emelkedni fog. Hasonlóan, amennyiben a téves riasztások aránya csökken, a rendszer érzékenysége is - és ezáltal az érzékelési arány is - csökkenni fog. Általános érvényű az a megállapítás is, hogy minél több idő áll az algoritmus rendelkezésére, hogy elemezze az adatokat, annál jobb eredményt tud nyújtani, viszont a nagyobb érzékelési idő nagyobb forgalomi hatást, súlyosabb következményeket is jelent. Más szóval: az TTD növelésével mind a DR mind a FAR javulni fog. Az érzékelési arány és a téves riasztási arány az algoritmus hatékonyságát írja le, míg a közepes érzékelési idő az alkalmazhatóságát. Ezen összefüggések miatt a legtöbb tanulmány több DR, FAR és TTD értékkel írja le az algoritmus teljesítményét, és ezért nem csupán egy optimális lehetőség létezik. Ezeket az értékeket minden telepítés előtt kalibrálni kell, megteremtve a még eltérhető téves riasztások számának és az elfogadható érzékelési időnek illetve aránynak az egyensúlyát, minden esetben az elérni kívánt célnak megfelelően.

Teljesítmény index

Az algoritmusok kalibrálása magában foglalja a különböző paraméter értékekkel történő tesztelést, mindaddig amíg az optimális értéket meg nem határoztuk. Az optimális érték megtalálása gyakran bonyolult mivel - mint láttuk - a paraméter értékek kapcsolatban állnak egymással. Az optimum-pont általában ott van, ahol az érzékelési arány növelése nem vezet a téves riasztási arány nagymértékű növekedéséhez. Kettőnél több paraméter kalibrálása esetén optimalizációs jártasságra van szükség. Ennél a folyamatnál a szakirodalom a TI - Teljesítmény

Indexet (PI - Performance Index) használja. A kisebb PI érték jelöli a jobb teljesítményt:

$$PI = \left[\frac{100 - DR}{100} \right]^m FAR^n MTTD^p,$$

ahol: $m > 0$, $n > 0$ és $p > 0$ ($DR < 100\%$, $FAR > 0\%$ és $MTTD > 0$ értékekre). Az m , n és p a fenti egyenlőségben az DR, FAR és MTTD paraméterek egyenkénti súlyozására szolgáló tényezők. Ezen tényezők tipikus értékei: $m = 1$, $n = 1$ és $p = 1$. Nagyobb értékek az adott jellemző kiemelt fontosságára utalnak.

1.2. Eseményérzékelő algoritmusok

Az elmúlt 30 évben négy fő típusát fejlesztették ki a számítógép-alapú algoritmusoknak:

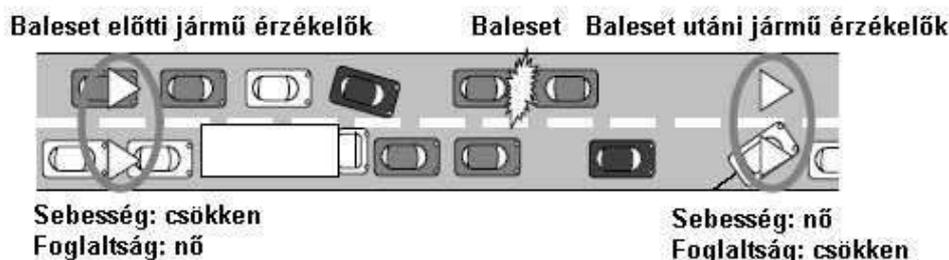
1. összehasonlító vagy minta felismerő,
2. forgalmi modell és elméleti algoritmusok,
3. statisztikai illetve idősor-alapú és
4. mesterséges intelligenciát alkalmazó más néven fejlődő módszerek.

Míg a minta felismerő illetve statisztikai algoritmusokat alkották meg először az 1970-es években, a mesterséges intelligenciát alkalmazóak a legújabbak. Ezen metódusok némelyikét videós képfeldolgozást alkalmazó eseményérzékelésnél is használják. Mivel ez a technológia egyik csoportba sem sorolható tisztán, ezért ezt külön fejezetben tárgyalom. Az új programok illetve technológiák a már kiépített eseményérzékelő algoritmusok közül többet is használnak döntést-támogató rendszereken keresztül. Például, öt algoritmus használatakor a döntést-támogató rendszer akkor jelezhet riasztást, amennyiben az ötből legalább három algoritmus "egyetért".

A hurokdetektorral történő érzékelő algoritmusokat két nagy csoportba lehet osztani: azok, amelyek egyetlen (detektor) állomás mérésein alapulnak (pl. McMaster algoritmus) és azok, amelyek összehasonlító eljárást használnak két - az autópálya mentén térben elkülönülő - (detektor) állomás adatainak kiértékeléséhez (pl. California algoritmusok). Az összehasonlító eljárás kedvezőbb mint az egyedi detektoros eljárás (mert ez utóbbi hajlamos túlzott mértékű téves riasztásokra), ellenben tény, hogy a detektor állomások közötti kommunikáción alapuló metódus növeli a költségeket és csökkentheti a megbízhatóságot.

1.2.1. Összehasonlító vagy minta felismerő algoritmusok

A minta-alapú algoritmusok a legelterjedtebbek a mindennapi használatban. Foglaltsággal, forgalomnagysággal dolgoznak, melyeket általában induktív hurokdetektorokkal gyűjtenek össze. A lehetséges balesetek azonosítása az adatok közötti azon minták felismerésével történik, amelyek nem tekinthetők "normálisnak" az út azon szakaszán. Azon a elven alapszik működésük, hogy baleset esetén a helyszín előtti hurokdetektor foglaltsága megnő, míg ezzel egy időben a baleseti helyszín utáni detektor foglaltsága lecsökken, melyet a következő egyszerűsített ábra szemléltet:



1.1. ábra. Forgalmi változók alakulása baleset esetén

Ennek felismeréséhez előre beállított határértékek szükségesek, melyek leírják a "normál áramlást". Bármilyen a "normál áramláson" kívülre esik, riasztást kell indítsón. A határértékek megfelelő beállítása bonyolult és időigényes feladat, azonban igen fontos a legjobb teljesítmény elérése érdekében.

A California Algoritmus

Egyike a legelsőként kifejlesztetteknek (az 1960-as évek végén a Los Angeles-i autópályára tervezték), a California Algoritmus - néha TSC 2 (Traffic Services Corporation - Forgalmi Szolgáltató Vállalat) algoritmusként is megnevezve - amely máig a legtöbb módszer viszonyítási alapjaként használt eljárás. Az algoritmus három tesztet használ események detektálásához, melyekhez az adatokat két szomszédos detektor foglaltság mérései szolgáltatják. Baleset lehetőségét akkor jelzi, amennyiben e három teszt eredményei túllépi az előre beállított határértékeket. E három vizsgálat a következő:

1. A menetirány szerinti térben előbb elhelyezkedő érzékelő foglaltságának (OCC_i) és a később elhelyezett (szomszédos) érzékelő foglaltságának (OCC_{i+1}) különbségét viszonyítja egy T_1 határértékhez. Amennyiben ezt túllépi a számított érték, a 2. pontra lép tovább.
2. A két érzékelő foglaltság-különbségének és a "felálló" irányban elhelyezett detektor foglaltságának hányadosát vagyis $\frac{OCC_i - OCC_{i+1}}{OCC_i}$ értékét viszonyítja egy T_2 határértékhez. Amennyiben ez meghaladja T_2 értékét az algoritmus a 3. pontra lép.
3. Ezúttal a két detektor foglaltság-különbségét a második járműérezékelő foglaltságával osztja, és az így kapott $\frac{OCC_i - OCC_{i+1}}{OCC_{i+1}}$ értéket a T_3 határértékhez viszonyítja. Abban az esetben ha ezt a határértéket is túllépték, baleset lehetőségét jelzi a rendszer és megismétli a 2. pontban leírt lépést, amennyiben a $\frac{OCC_i - OCC_{i+1}}{OCC_i} > T_2$ reláció továbbra is fennáll az algoritmus balesetet jelez. Egy baleseti állapot akkor ér véget, amikor a T_2 határértéket már nem lépi túl az adatokból számított hányados.

A különböző határértékek beállítása tapasztalati adatokból történik. Bár a California Algoritmus egyszerű, mégis nehéz számításokat igényel a határértékek meghatározásához, amit minden egyes telepítés helyszínén el kell végezni, továbbá nagy hálózatok esetén külön határértékeket kell számolni a különböző útgeometriákhoz (pl.: felhajtók, összekapcsolódó részek, dombok, stb.). Egyszerű természetéből fakadóan teljesítménye nem olyan jó, mint a későbbi

javított kiadások vagy új technológiák esetében, de összességében egy megbízható algoritmus, melyet sok helyen használnak.

1.2.2. *Forgalom modell és Elméleti Algoritmusok*

Ezek az algoritmusok komplex forgalmi elméleteket használnak a forgalom viselkedésének leírására és előrejelzésére baleseti körülmények esetén. Az aktuális forgalmi paramétereket a modell által előre jelzett értékekkel hasonlítják össze.

Dinamikus Algoritmus

Ez az algoritmus az alapvető sebesség-sűrűség illetve forgalomnagyság-sűrűség összefüggéseket egyesíti az érzékelési elemzésben. Az algoritmus azt feltételezi, hogy a hirtelen változások az autópályán előre látható mintákat követnek. Két statisztikai hipotézis teszteli a folyamatokat, melyek a megfigyelt forgalmi adatokban vizsgálják az forgalomnagyság-sűrűség összefüggéseket: az MM (Multiple Model: Összetett Modell) eljárás és a GLR (Generalized Likelihood Ratio - Általános Valószínűségi Arány) eljárás. Az algoritmus az MM eljárást használja a feltételes valószínűségek megállapítására, melyeket mint vezérlő mértékeket hasznosít események azonosításakor. A GLR metódus méri annak valószínűségét, hogy a megfigyelt forgalomnagyság-sűrűség minta mennyire utal balesetre. Az algoritmus a forgalmi áramlás dinamikus modellén alapul, amely a forgalom mind makroszkopikus mind mikroszkopikus szemléletmódját magában foglalja a sebesség-sűrűség kapcsolatának leírásához. Mivel a detektorok fix-pontokról szolgáltatnak foglaltság értékeket, ezért az időbeli átlagokat először térbeli átlaggá kell konvertálni (tér-idő szemlélet).

Katasztrófa Elmélet - A (módosított) McMaster Algoritmus

A Katasztrófa Elmélet nevét a hirtelen diszkrét változásokról kapta, melyek egy megfigyelt változóban következnek be, míg más kapcsolódó változók sima és folytonos átmenetet mutatnak. Ezek a változók a sebesség, forgalomnagyság és foglaltság. Amennyiben a sebesség nagymértékben lecsökken a foglaltság illetve forgalomnagyság megfelelő növekedése nélkül, a riasztás megszólal. Ebben a tekintetben a Katasztrófa Elmélet alapú algoritmusok képesek különbséget tenni a balesetek és a visszatérő torlódások között. A torlódások lassan épülnek fel, amíg a balesetek hirtelen sorfelépülést és drasztikus változásokat okoznak a sebességben. Az algoritmus ezt a jelenséget használja ki. A különbség a Katasztrófa Elmélet és a minta-alapú algoritmusok között az, hogy ez utóbbi eljárások egyedi változón és előre beállított határértékeken alapul, míg a Katasztrófa Elmélet összetett változókat használ és ezeket hasonlítja össze a visszatérő torlódások korábbi adatainak trendjeivel. Az egyetlen algoritmus amelyik ebbe a csoportba tartozik a McMaster Algoritmus. Az algoritmus függvényei egyetlen detektorállomás adatait használják. A korábbi adatokat felhasználva forgalomnagyság-foglaltság illetve sebesség-foglaltság diagramokat készítettek. Az algoritmus logika két összehasonlító tesztet hajt végre a diagramok vázolta minták és az aktuális hurokdetektor értékek között. Az első vizsgálat azt figyeli, hogy a forgalom a detektor helyén zsúfolt, torlódott-e. Amennyiben torlódást érzékel, az algoritmus megpróbálja meghatározni az eredetét a forgalmi állapot vizsgálatával a térben később elhelyezkedő detektornál.

1.2.3. Statisztikai illetve idősor alapú algoritmusok

A statisztikai metódusok a "real-time" forgalmi adatokat hasonlítják össze az előrejelzésekkel. Az algoritmusok modellezik az aktuális forgalmi mintákat és idősorok segítségével előrejelzik a várható értékek tartományát. Az előrejelzésekhez képest bármilyen nem várt változás a forgalomban eseménynek tekinthető. A módszer előnye, hogy nagy mennyiségű adat beszerzése szükséges mielőtt az algoritmus végrehajtódik és ezáltal a téves riasztások száma csökkenthető.

Statisztikai algoritmusok

Ezek az algoritmusok hagyományos statisztikai technikák használatával határozzák meg, hogy a vizsgált detektor adatok jelentősen eltérnek-e a becsült vagy előre jelzett értékektől.

Bayesian Algoritmus. Az algoritmus, hasonlóan a California-alapú eljárásokkal, két detektor közötti foglaltság érték különbségével dolgozik, de ezután a Bayesian statisztikai különbségeket használja fel annak valószínűségének számítására, hogy a nagy különbség a foglaltság értékek között baleset vagy visszatérő torlódás következtében alakult ki. Három adatbázis szükséges ehhez: foglaltság és forgalomnagyság értékek baleseti körülmények esetén, foglaltság és forgalomnagyság értékek balesetmentes körülményekre továbbá a balesetek típusáról, helyzetéről valamint súlyosságáról szóló adatbázis. Ezen korábbi adatok elemzésével felállítható az a statisztikai tartomány amely alapján eldönthető hogy az adott foglaltság-különbséget baleset okozta vagy sem. Más algoritmusokhoz képest a Bayesian nem "igen" vagy "nem" válaszokat nyújt, hanem százalékos értékben fejezi ki a baleset valószínűségét. Így az algoritmus akkor riaszt, amikor az erre a valószínűsége felállított határértéket átlépi a vizsgálat eredménye. A rendelkezésre álló "off-line" teszteredmények jó érzékelési arányt és téves riasztási arányt (100% illetve 0%) mutattak, viszont az érzékelési idő viszonylag magas (négy perc) volt. Egy tanulmány kimutatta, hogy a Bayesian-alapú PNN (Probabilistic Neural Network - Valószínűségi Neurális Hálózat) módosított formája amely a statisztikai különbség megközelítését használja, képes megvalósítani ezen általános követelményeket.

Idősor alapú és simító/szűrő algoritmusok

Ezen algoritmusok hosszútávon elemzik vagy simítják az alapadatokat ezzel kizárva a forgalom olyan rövid ideig tartó zavarásait mint véletlenszerű ingadozások, forgalmi lüktetések illetve sűrűsödési hullámok. A feldolgozott adatokat általában előre meghatározott határértékekkel hasonlítják össze.

Idősor ARIMA Algoritmus. Az ARIMA modellek rövid-távú előrejelzéseket nyújtanak a foglaltságról és a kapcsolódó 95 százalékos megbízhatósági intervallumokról. Ezek az alacsonyrendű lineáris modellek három korábbi intervallum megfigyelt adatai alapján jelzik előre a körülményeket. Amennyiben a jelenlegi jellemzők az előre jelzett tartományon kívül esnek a riasztás megszólal. Egy vizsgálat során az algoritmust a California algoritmussal hasonlították össze. A téves riasztási arányt egy százalékra állították be mindkettőnél, ekkor az ARIMA mind érzékelési arány mind érzékelési idő tekintetében felülmúlta a California Algoritmust. Az

algoritmus közepes és nagy forgalom esetén megfelelően teljesít, de megkérdőjelezhetően produkál alacsony forgalom nagyság mellett. Továbbá kissé egyszerűsített modellezési jellemzői miatt nem képes magyarázatot nyújtani számos a közlekedés természetéből fakadó összetettségre, hisz az algoritmus mindennapos stabilitást feltételez, továbbá paramétereit csupán egy típusú adatból határozzák meg. Az olyan jelenségek mint a forgalom heti, napi (órai) változásai, időjárási körülmények, különböző kivitelezések illetve más változók mind rontják az algoritmus teljesítményét. Épp emiatt a korlátozások miatt nem használatos, további fejlesztése sem történt meg.

1.2.4. Mesterséges Intelligencia használata eseményérzékeléshez

A mesterséges intelligencia automatikus eseményérzékelésre történő használata jelenleg is kutatási terület. Ezek az algoritmusok vagy szabály-alapúak vagy "megtanulják" felismerni a balesetekre jellemző mintákat.

Neurális Hálózatok

Az 1990-es évek elején mutatták ki a mesterséges Neurális Hálózatok (Neural Network - Ideghálózat) alkalmazhatóságát automatikus eseményérzékelésre. A neurális hálózat egy olyan algoritmus ami az emberi agy idegi felépítését modellezi. Az információ több párhuzamos csatornán van szétszétva a feldolgozó egységek más néven neuronok között. Minden egyes neuron összeköttetésben áll bizonyos számú szomszédjával, ahol változó összeköttetési együtthatóval vesz részt a kapcsolatban, amely a kapcsolat erősségét reprezentálja. A tanulási folyamat úgy zajlik, hogy a kapcsolatok erősségét változtatjuk olyan irányba, ami a teljes rendszert a helyes eredmény elérésére sarkallja. Az automatikus eseményérzékelő neuronhálózatok általában többrétegűek és "feed-forward"¹ struktúrájúak. Az ilyen struktúrák három réteget tartalmaznak, bemeneti réteg, ami fogadja az információkat a hurokdetektoroktól, középső réteg, ami feldolgozza ezeket az adatokat és a kimeneti réteg, amely baleseti vagy balesetmentes jelzést ad. A hálózat begyakoroltatását üzembeállítása előtt el kell végezni. Ezen próbák és hibák alapján a hálózat megtanulja a helyes fontossági súlyokat használni a be- és kimeneteknél. A tanulási módszer rendszerint a következő sémák valamelyike. Felügyelt tanulás során a kiinduló kapcsolat a feldolgozó (rejtett) réteg neuronjai között véletlen szervezésű, amit újraszerveznek, amint a rendszernek "megmondják" milyen közel áll a probléma megoldásához. Nem felügyelt tanulás esetén a rejtett neuronoknak meg kell találniuk a módot az önszervezésre külső segítség nélkül. A 90-es években kezdődő kutatások eleinte igen kedvezőek voltak mindhárom teljesítménymérő paraméter esetében. Az Neurális Hálózat algoritmus 85 százalékos érzékelési arányt és 0,075 százalékos téves riasztási arányt mutatott három perces érzékelési idővel. Ez a 0,075 százalékos arány azt jelenti, hogy mintegy 11 óránként riaszt hibásan a rendszer (800 detektor esetén). Azonban a Neurális Hálózat még csak kezdeti stádiumban van, további fejlesztésekkel és aktuális eredmények felhasználásával jó megoldást nyújtana ez az algoritmus.

¹ Feed-forward ("Add tovább") kapcsolat: Az első réteg neuronjai a kimenő adataikat továbbküldik a következő réteg számára, de semmiféle visszacsatolást a második rétegtől nem kapnak.

1.3. Videokamerák felhasználása eseményérzékelésre

A videós érzékelő rendszerek számos előnnyel rendelkeznek melyek a nagy térbeli érzékelési képességükből fakadnak hiszen egy kamerával több sáv is megfigyelhető. Ez lehetővé teszi a lökéshullámok illetve egyéb térbeli forgalmi paraméterek érzékelését (mint például: sűrűség, sorhossz vagy sebesség az út adott szakaszán) melyeket a hagyományos eszközökkel nem lehet egyszerűen meghatározni. Mindemellett a videós érzékelés olyan másodlagos információkat is nyújt, mint például forgalom a leállósávban, megállt járművek, sávváltások, sebességkülönbségek vagy forgalom lassulás az ellenkező irányban. Különbséget kell tennünk az olyan videós rendszerek között melyek képesek jelezni, eseményt érzékelni, sorhosszt vagy más paramétereket meghatározni minimális emberi beavatkozással vagy nélkül, és a jelenleg elterjedt "manuális" felügyelettel működők között. A most következőkben az automatizált rendszereket mutatom be.

Videós automatikus eseményérzékelő rendszereket általánosan három kategóriába oszthatjuk. Az ennél részletesebb osztályozás bonyolult, hisz a jelenleg működő rendszerek legtöbbször hibrid formában működik.

1. **"Tripwire" rendszerek:** Az ilyen rendszerekben a kamera virtuális detektorokkal működik, melyeket a képen elhelyezett kis területek valósítanak meg. Több detektor is elhelyezhető a képen, és ezeket könnyen az út geometriájához igazíthatjuk. Ennek a kialakításnak hátránya, hogy ebben az esetben az eseményérzékelési algoritmus a már megismert eljárásokkal azonos, továbbá a detektorok pontossága erősen függ a kamera látóterétől.
2. **Követő rendszerek:** Ezek a rendszerek érzékelik a kamera képén áthaladó járműveket és követik azokat. Ezáltal a járművek mozgásának mikroszkopikus leírását kapjuk, mely olyan adatokat tárhat fel, mint hirtelen sávváltások, forgalommal ellentétes irányban haladó járművek illetve megállt járművek. Ez növeli a szükséges számítási teljesítményt illetve működéséhez jól felismerhető járművek szükségesek, továbbá még korlátozottabb kamera elhelyezést igényelhet.
3. **Térbeli elemző rendszerek:** A harmadik kategória a videókép által szolgáltatott két-dimenziós információk elemzésére koncentrál. A forgalom járművenkénti figyelembe vétele mellett a rendszer azt próbálja meg leírni, hogy a látható útfelület hogyan van kihasználva egy adott időpillanatban. Ennek változásainak elemzésével határozhatjuk meg a forgalmat zavaró eseményeket. Az útfelület kihasználása három csoportba osztható: nincs forgalom, mozgó forgalom és álló forgalom [4].

A fejezet [1], [2], [3] és [4] felhasználásával készült.

2. AZ AUTÓPÁLYA FORGALOM DINAMIKUS MODELLJE

A fejezet tartalmát tekintve három részre bontható. Az elsőben az elméleti alapokról lesz szó. Ez a szakasz az autópálya forgalom modellezéséről, a felhasznált forgalomtechnikai eredményekről és a levezetett modellegyenletekről szól. Az egyenletekben szereplő tagokról, azok tartalmáról illetve a modellben szereplő paraméterekről és jelentésükről szemléletes képet alkothatunk.

A második részben a valós adatokról, a valós helyszínről írok, vagyis arról, hogy a fejezet elején levezetett elméleti alapokat hogyan valósítottam meg gyakorlatban. Bemutatom a rendelkezésemre álló forgalmi adatok feldolgozásának módját és az így kapott információk felhasználását. Ismertetem a modell realizációját MATLAB SIMULINK környezetben, a felépítés fontosabb pontjaira részletesen kitérek. A megfelelő modell érdekében történt validáció gondolatmenetéről, az egyes paraméterek "off-line" kalibrációjáról is írok.

Végül bemutatom az így kapott eredményeket és értékelem a felállított modellt.

2.1. Modell egyenletek levezetése

A forgalom megmaradásának alapegyenlete

A forgalom makroszkopikus leírásához jól definiált változókra van szükségünk, melyek segítségével a járművek átlagos viselkedését leírhatjuk. Jól ismert, hogy a $\rho(x, t)$ forgalomsűrűséget az egység hosszra eső járművek számával $\left[\frac{jm}{km}\right]$, a $q(x, t)$ forgalom nagyságot az időegység alatt áthaladó járművek számával $\left[\frac{jm}{h}\right]$ definiáljuk, míg $v(x, t)$ a járművek átlagsebességét jelöli $\left[\frac{km}{h}\right]$. Ezeket a változókat felhasználva az autópálya forgalmat - makroszkopikus szemléletmódban - olyan folyadéknak tekinthetjük, mely jellemzői a forgalomsűrűség: $\rho(x, t)$ és a forgalom nagyság: $q(x, t)$. Az áramlástan ismeretével analóg módon felírhatjuk a forgalom megmaradási alapegyenletét:

$$\frac{\partial \rho}{\partial t} + \frac{\partial q}{\partial x} = r - s \quad (2.1)$$

ahol r illetve s jelöli a fel- illetve lehajtási pontokat, áramlástan szóhasználatnál maradva: a forrást és a nyelőt. A könnyebb kezelhetőség kedvéért a parciális differenciálegyenletekből közöséges differenciálegyenleteket képzünk. Ennek érdekében az autópályát N darab ún. szegmensre osztjuk fel, melyek hossza egyenként Δ_i , $i = 1, \dots, N$ továbbá megengedjük, hogy az egyes szegmensek rendelkezzenek fel- illetve lehajtó sávval. Ezek után bevezetjük a következő, térben diszkrét forgalmi változókat:

$\rho_i(t)$: az i -edik szegmensben tartózkodó járművek száma osztva a szegmens hosszával (vagyis $\Delta_i - vel$).

$v_i(t)$: az i -edik szegmens járműveinek átlagsebessége.

$q_i(t)$: időegység alatt az i -edik szegmenst elhagyó járművek száma.

$r_i(t), s_i(t)$: az i -edik szegmens felhajtó illetve lehajtó járműveinek száma (amennyiben létezik).

Ezek után a most bevezetésre került változók segítségével a megmaradási egyenletet a következő térben diszkrét alakban írhatjuk fel [8]:

$$\dot{\rho}_i(t) = \frac{1}{\Delta_i} [q_{i-1}(t) - q_i(t) + r_i(t) - s_i(t)]. \quad (2.2)$$

Az alapegyenlet kiterjesztése a sebesség-sűrűség karakterisztikával

Homogén forgalom esetén az áramlástanból ismert:

$$q = \rho \cdot v \quad (2.3)$$

egyenlőség áll fenn. Azonban a forgalom speciális tulajdonsága, mely semmilyen más folyadék (illetve közeg) áramlásához nem hasonlítható, hogy a forgalomsűrűség növekedésével a sebesség folyamatosan csökken. A jelenség matematikai leírására a következő összefüggés áll rendelkezésünkre [8]:

$$V(\rho) = v_{free} \cdot \exp \left[-\frac{1}{a_m} \left(\frac{\rho}{\rho_{cr}} \right)^{a_m} \right], \quad (2.4)$$

ahol:

v_{free} : a szabad áramlási sebesség,

ρ_{cr} : a kritikus forgalomsűrűség,

a_m : pedig konstans paraméter.

Egyszerűsített dinamikai modell

Az eddigi eredményeink már tartalmaznak olyan forgalomtechnikai eredményeket, melyek a forgalomlefolrás egyediségét modellezik. Ezek után a forgalom másik fontos tulajdonságát, a dinamikát is modellezhetjük. Ezt a dinamikát egy τ_i időeltolással vehetjük figyelembe, mely tartalmát tekintve a szegmensbe belépő forgalom τ_i idővel való eltolását jelenti. Egyértelmű, hogy a kérdéses τ_i paraméter az adott szegmens megtételéhez szükséges átlagos utazási időt jelenti, és ezáltal értéke a fentebb vázolt sebesség-sűrűség görbétől függ [8].

A sebességre vonatkozó egyenlet bevezetése

Az eddigi észrevételeket felhasználva az i -edik szegmens átlagsebességére a következő differenciálegyenletet kapjuk:

$$\begin{aligned} \dot{v}_i &= \frac{1}{\tau} (v(\rho) - v_i) \\ &+ \frac{v_i}{\Delta_i} (v_{i-1} - v_i) \\ &- \frac{\nu}{\tau \Delta_i} \frac{\rho_{i+1} - \rho_i}{\rho_i} \end{aligned} \quad (2.5)$$

ahol: ν konstans paraméter.

Az egyenlet megfelel a forgalom áramlás makroszkopikus leírásának. Az egyenlet jobb oldalának első tagja az ún. relaxációs tag, mely tartalmazza a sebesség-sűrűség karakterisztikát, mint egy "ideális" érték az aktuális ρ_i mellett. A második tag, az áramlást fejezi ki, az i -edik szegmensbe érkező járművek hatását. A harmadik tag pedig a várakozást fejezi ki, a szegmenst követő forgalmi viszonyok hatását a szegmens forgalmára [8].

Térben és időben diszkrét egyenletek

A már meglévő egyenleteinkben a folytonos időről a következőkben áttérünk diszkrét időre. Ezt a $t = Tk$, $k = 0, 1, 2, \dots, K$ időköz bevezetésével végezhetjük el, ezután a forgalmi változók időben diszkrét értékei ($\rho_i(k)$, $v_i(k)$ illetve $q_i(k)$) az egyes k időpillanatokra vonatkoznak. Ezt követően a diszkrét egyenletek a 2.2, 2.5, 2.3 folytonos egyenletekből egy egyszerű Euler-formula felhasználásával származtathatóak:

$$\begin{aligned} \rho_i(k+1) &= \rho_i(k) + \frac{T}{\Delta_i n} [q_{i-1}(k) - q_i(k) + r_i(k) - s_i(k)] \\ v_i(k+1) &= v_i(k) \\ &+ \frac{T}{\tau} [V(\rho_i(k)) - v_i(k)] \\ &+ \frac{T}{\Delta_i} v_i(k) [v_{i-1}(k) - v_i(k)] \\ &- \frac{\nu T}{\tau \Delta_i} \frac{\rho_{i+1} - \rho_i(k)}{\rho_i(k) + \kappa} \end{aligned} \quad (2.6)$$

$$q_i(k) = \rho_i(k) \cdot v_i(k) \cdot n \quad (2.7)$$

ahol, mint látható egy további κ paramétert adtunk az egyenletekhez, ezzel növelve a modell pontosságát alacsony forgalomsűrűség esetén. Továbbá n jelöli a szegmensben található sávok számát.

A most levezetett egyenletek azonban csak viszonylag kis szegmenshossz esetén írják le pontosan a forgalom lefolyását. Az irodalomban talált értékek alapján a dinamikus modell, másnéven METANET modell 500 méter körüli szegmenshossz esetén megfelelő. Ahhoz, hogy a teljes vizsgálni kívánt autópálya szakaszt modellezni tudjuk először szegmensekre osztjuk a szakaszt, majd a szegmenseket összekapcsoljuk az ún. határfeltételekkel (határ változókkal), melyek a következők:

az i -edik szegmensből kilépő forgalom nagyság (q_i) a következő szegmens belépő forgalmát jelenti (megmaradási egyenlet)

az i -edik szegmens átlagsebességét a következő $i + 1$ -edik szegmens számára a sebességre vonatkozó egyenletben, az áramlási tagban jelenik meg (v_{i-1})

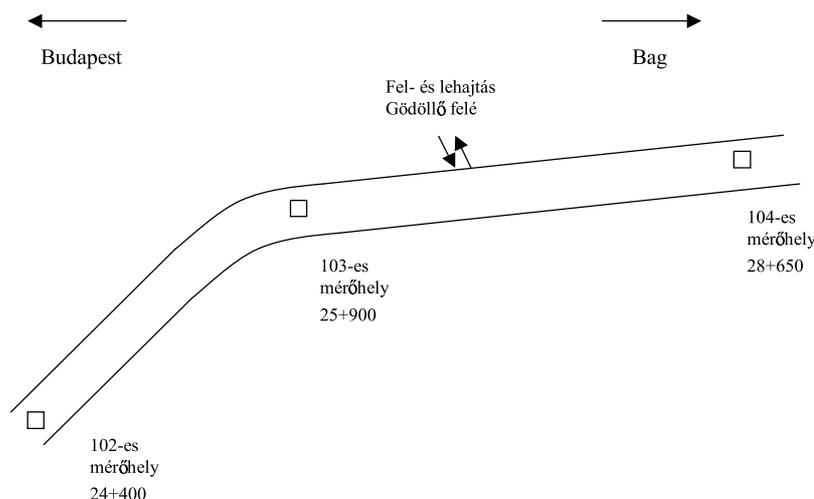
az i -edik szegmens forgalomsűrűségének értékét az előző $i - 1$ -edik szegmens számára a sebességre vonatkozó egyenletben jelenik meg, a várakozást leíró tagban (ρ_{i+1})

Ezek a határfeltételek kapcsolják egymáshoz a szegmenseket, mely alapján tetszőleges hosszúságú autópálya-szakasz felépíthető körülbelül 500 méter hosszú szegmensekből. Az egyenleteket és határfeltételeket jobban megvizsgálva láthatjuk, hogy az első szegmens számára szükséges a v_{i-1} érték amely ilyen formában ellentmondáshoz vezet, ennek feloldásának érdekében a $v_0 = v_1$ helyettesítést alkalmaztam. Hasonlóan az utolsó szegmens számára szükséges a ρ_{i+1} értéke amely esetében a $\rho_{N+1} = \frac{q_N}{v_N n}$ helyettesítést alkalmaztam [8].

2.2. Valós adatok, a vizsgált helyszín

Helyszínrajz

Az Állami Autópálya Rt. által nyújtott adatok az M3-as autópálya Gödöllő melletti szakaszáról valók. A 102-es, 103-as és a 104-es mérőhelyek 1-es és 2-es, vagyis Budapest felé haladó sávjainak forgalmi adatai álltak rendelkezésemre. A mérőhelyek elrendezését a következő egyszerűsített ábra szemlélteti:



2.1. ábra. Helyszínrajz, az M3-as autópálya Gödöllő melletti szakasza

A detektorok által gyűjtött adatok¹

Az autópályán elhelyezett detektorok szolgáltatnak adatokat a forgalom lefolyásáról. Az autópálya egy adott keresztmetszetén 4 darab detektor van elhelyezve, a négy sávnak megfelelően. Egy ilyen keresztmetszetet nevezünk mérőhelynek, a központi számítógépbe mérőhelyenként érkeznek az adatok, melyek tehát az adott keresztmetszet négy sávjának adatait tartalmazzák. A detektor az egyes elhaladt járművek adatait rögzíti, a következő formátumban:

1. Jármű áthaladásának időpontja. Ennek rögzítése negyedórás blokkokban történik. Formátuma: *óópp*, ahol a perc értékei 00,15,30 illetve 45 lehetnek. A detektor kiértékelés ma már nem negyedórás blokkok szerint történik, a módosított rendszerben a perc értékek közvetlenül elérhetőek. Tehát a 1215 jelöli 12 óra 15 percet, a későbbi verziókban pedig már a 1216 is lehetséges.
2. A mérőhely háromjegyű azonosítója. A sorszámozás Budapesttől kezdődik. Az első mérőhely azonosítója: 100, a következő 101 és így tovább.
3. Sáv azonosítója, lehetséges értékei:
 - 1 - Budapest felé haladó szélső sáv.
 - 2 - Budapest felé haladó belső sáv.
 - 3 - Budapesttől távolodó belső sáv.
 - 4 - Budapesttől távolodó szélső sáv.

¹ Forrás: Állami Autópályakezelő Részvénytársaság

4. Az előző jármű elhaladásától eltelt idő, vagyis a követési idő, másodpercben megadva. Ennek segítségével a tizenöt- illetve egyperces blokkokban is azonosíthatóak a járművek, azok pontos elhaladásának időpontja.
5. Hurok foglaltsági idő (FOGL), ezred másodpercben megadva.
6. Hurok áthaladási idő (ATH), ezred másodpercben megadva.
7. Az elhaladó jármű pillanatnyi sebessége. Ez a következő módon számított érték:

$$v = \frac{18000}{ATH}.$$

8. A jármű hossza mely szintén számított érték:

$$L = \frac{\text{int} \left[\left(5 \frac{FOGL}{ATH} - 1.2 \right) 10 + 0.5 \right]}{10}.$$

2.2.1. Az adatok beolvasása

Az Állami Autópálya Rt.-től kapott mérőállomás-adatokat a rendszer működéséhez MATLAB környezetben használtam fel. A fent vázolt formátumban (txt-fileban) kapott adatok beolvasását a következő módon valósítottam meg.

A fájl megnyitása után az időblokkokból másodperc értékeket állítottam elő. Ezután a modell számára szükséges adatokat tartottam már csak meg, vagyis az elhaladás időpontját, követési időt, a sáv számát és a pillanatnyi sebesség értékeket. Következő lépésben a mérőhely adatait sávok szerint bontottam szét. Majd az egyes sávok adataiban a követési idők kummulált összegét vettem. Ezt az értéket a már előállított másodperc értékekhez adva kaptam meg az időben folytonosan számított járműelhaladási időpontokat. Ezen lépéseket a A.1 MATLAB file tartalmazza.

Az adatok további feldolgozása a következők szerint történt. A rendelkezésre álló adatokat sávok szerint vizsgáltam és a modellben használt szimulációs lépésköz szerint bontottam fel. Ez az érték 10 másodperc. Ennél a ciklusnál az adott 10 másodperces blokkban érkező járművek számát összeadtam, míg a sebességértékeket a blokkban elhaladó járművek harmonikus sebességével helyettesítettem. Erre azért van szükség, mivel a detektorokból nyerhető sebesség adatok időbeliek, a modellhez azonban térbeli átlagsebességre van szükség, amelyet az időbeli sebességek harmonikus átlagaként kaphatunk. Ezután a kapott értékeket simítottam a feldolgozás érdekében. Ehhez kilencedrendű mozgó átlag szűrést alkalmaztam. Az adott mérőállomás két sávjának eképpen feldolgozása után mérőhelyi adatokat állítottam elő. A két sávban érkező járművek számát összeadtam, sebességüket pedig ismét átlagoltam. Az így előállt adatsort végül mégegyszer simítottam. Az eljárást az A.3 MATLAB file tartalmazza.

2.3. Modellparaméterek "off-line" kalibrációja

Mint azt láthattuk az autópálya forgalom dinamikus modellje számos ismeretlen paramétert tartalmaz, melyek értékeit számos tényező befolyásolja, mint pl.: útgeometria, időjárási körülmények, emberi tényezők. Az ismeretlen paraméterek a következők:

v_{free} : szabad áramlás sebessége, $\left[\frac{km}{h}\right]$

a_m : modellparaméter,

ρ_{cr} : kritikus forgalomsűrűség, $\left[\frac{j}{km}\right]$

τ : szegmensen belüli átlagos utazási idő, [h]

κ : paraméter, $\left[\frac{j}{km}\right]$

ν : paraméter, $\left[\frac{km^2}{h}\right]$

Ezen paraméterek beállítását először "off-line" végeztem el, vagyis a már rendelkezésre álló adatokból határoztam meg a megfelelő értéket. A három legfontosabb paraméter, melyek értékeire a modell a legérzékenyebb, a következők: v_{free} , ρ_{cr} és a_m .

2.3.1. Szabad áramlási sebesség, kritikus forgalomsűrűség, modellparaméter meghatározása

Az adatokat a már ismert eljárással olvastam be. Negyedórás ciklusokkal dolgoztam, a következő lépések szerint számoltam [16]:

1. A ciklusban elhaladt járművek száma:

$$N = \sum n, (db).$$

2. A ciklusban elhaladt járművek harmonikus átlagsebessége:

$$v_s = \frac{N}{\sum_{i=1}^N \frac{1}{v_i}}, \left(\frac{km}{h}\right)$$

3. Hurok foglaltsági idők összegzése:

$$f = \sum_{i=1}^N FOGL, (ms)$$

4. Foglaltság:

$$F = \frac{f}{15 \cdot 60} \cdot 100(\%)$$

5. Relatív foglaltság:

$$L = \sum_{i=1}^N \frac{L_i}{3.6 v_i}, (s)$$

ahol L_i jelöli a jármű hosszát.

6.

$$M = \sum_{i=1}^N \frac{1}{v_i}, \left(\frac{m}{s}\right)$$

7. Effektív járműhossz:

$$L_{eff} = \frac{L}{M}, (m)$$

8. Maximális forgalomsűrűség:

$$\max \rho = \frac{1000}{L_{eff}}, \left(\frac{jm}{km}\right)$$

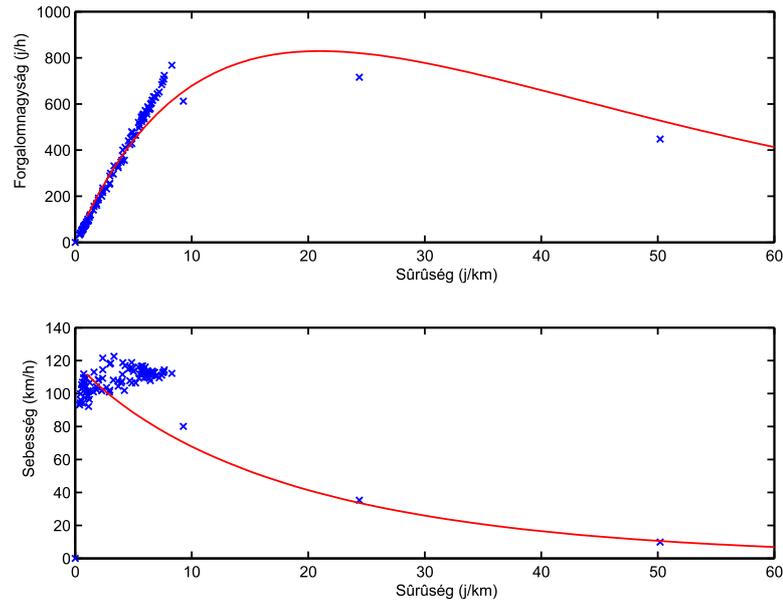
9. Aktuális forgalomsűrűség:

$$\rho = \frac{\max \rho \cdot F}{100}, \left(\frac{jm}{km}\right)$$

Az adatok feldolgozása után a kapott sűrűség-sebesség illetve sűrűség-forgalomnagyság párokat diagramban ábrázoltam. A pontpárookra görbét illesztettem, mely egyenlete:

$$V(\rho) = 120 \cdot \exp \left[-\frac{1}{0.9} \left(\frac{\rho}{21} \right)^{0.9} \right] \quad (2.8)$$

A következő ábra mutatja az így kapott sűrűség-forgalomnagyság illetve a sűrűség-sebesség diagrammokat:



2.2. ábra. Forgalmossűrűség-forgalomnagyság, forgalmossűrűség-sebesség

Az illesztett görbe egyenletét és a 2.4 egyenletet összevetve láthatjuk, hogy a modell három paramétereire kaptunk értéket:

$$\text{Kritikus forgalmossűrűség: } \rho_{cr} = 21 \left(\frac{j}{km} \right)$$

$$\text{Szabad áramlás sebessége: } v_{free} = 120 \left(\frac{km}{h} \right)$$

$$\text{Modell paraméter: } a_m = 0.9$$

A feladat megvalósítására írt MATLAB script a A.2-ban található.

2.3.2. Modell validáció

A modell ismeretlen paramétereit közül tehát hármat számolással határoztam meg, a fennmaradó paramétereket a következő módon "hangoltam" be.

A B.1 ábra mutatja a SIMULINK környezetben felépített modellezendő autópálya szakaszt. Mint látható a 4,25 kilométeres szakaszt nyolc darab szegmensre osztottam fel. A 104-es mérőhelytől a 103-as mérőhelyig elhelyezkedő 2,75 kilométeres szakaszt öt darab, egyenként 550 méter hosszú szegmensre ($\Delta_1 = 0.55$), míg a 103-as mérőhelytől a 102-es mérőhelyig fekvő 1,5 kilométeres szakaszt három, egyenként 500 méter hosszúságúra ($\Delta_2 = 0.5$). Az egyes szegmensek SIMULINK-ben felépített blokkvázlatát a B.2 ábra szemlélteti².

² Felhajtó járművek csak a 4. szegmensben vannak, a többi szegmens esetén értékük 0.

Ezek után az ismeretlen paraméterek közül a τ értékére 0,004h-t állítottam be, mely 100 $\left(\frac{km}{h}\right)$ -ás átlagsebesség mellett a szegmensek megtételéhez szükséges átlagos idő.

Ahogy a helyszínrajzon is látható, a 103-as mérőhely és a 104-es mérőhely közötti szakaszon található egy fel- illetve lehajtó sáv. Mivel ezek forgalmáról pontos értékek nem álltak rendelkezésemre, egyszerűsített feltételezéssel csak a felhajtó járművek hatását vizsgáltam. A vizsgált időintervallumban a 104-es mérőhelyen belépő illetve a 103-as mérőhelyen kilépő járművek számát egymáshoz viszonyítva ugyanis azt kaptam, miszerint több jármű lépett ki a 103-as mérőhelyen, mint amennyi belépett a 104-es mérőállomásnál. A különbséget órára levetítve körülbelül 50 járművet kaptam. A továbbiakban ezzel az értékkel, mint 50-es várható értékű, 15-ös szórású véletlen számmal modelleztem a felhajtó járműforgalmat.

A szimulációs lépésköz értéket $T = 10s$ -nak választottam (az adatok feldolgozása is eszerint történt, lásd A.3, mely közepes- és nagy forgalomnagyság mellett megfelelő. Ez az érték mutatja, hogy mekkora mintavételezési idővel olvassuk ki a mérőállomás adatait. Alacsony forgalomnagyság esetén ezen lépésköz értékének növelése célszerű, hiszen a járművek térbeli átlagsebességének számításakor az érték nagyon lecsökkenne.

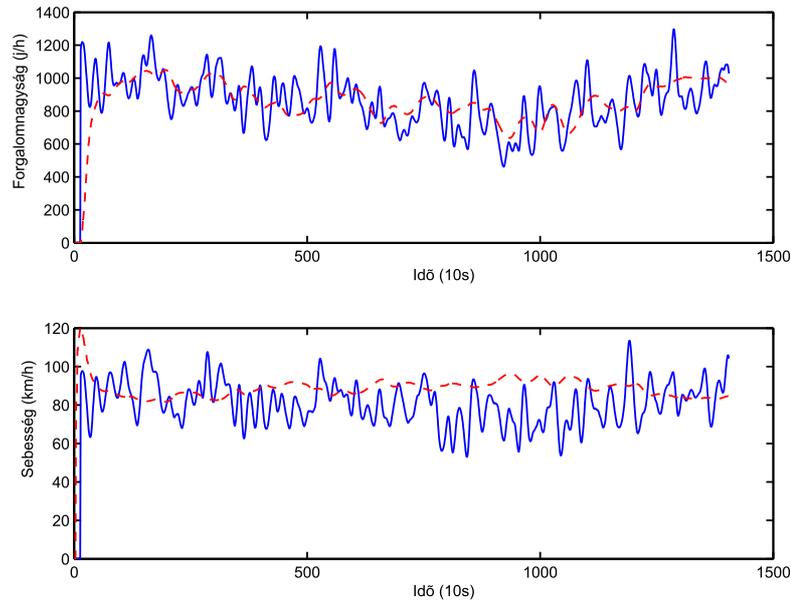
A modell validációja a következők szerint történt. A szakasz kezdő- (104) illetve végpontját (102) tekintettem bemenetnek, míg a modell kimenetének a két végpont között elhelyezkedő (103) mérőhelyet. A rendelkezésre álló adatsort tehát a szakasz végpontjain a modellnek folyamatosan beolvastam. A középső ponton - nevezzük ellenőrző pontnak - pedig összehasonlítottam a valós- illetve a modell által számított adatokat. Az egyes paraméterek hatását vizsgálva a következő értékek mellett adódott az optimális eredmény:

$$\kappa = 20 \left[\frac{j}{km} \right],$$

$$\nu = 22 \left[\frac{km^2}{h} \right].$$

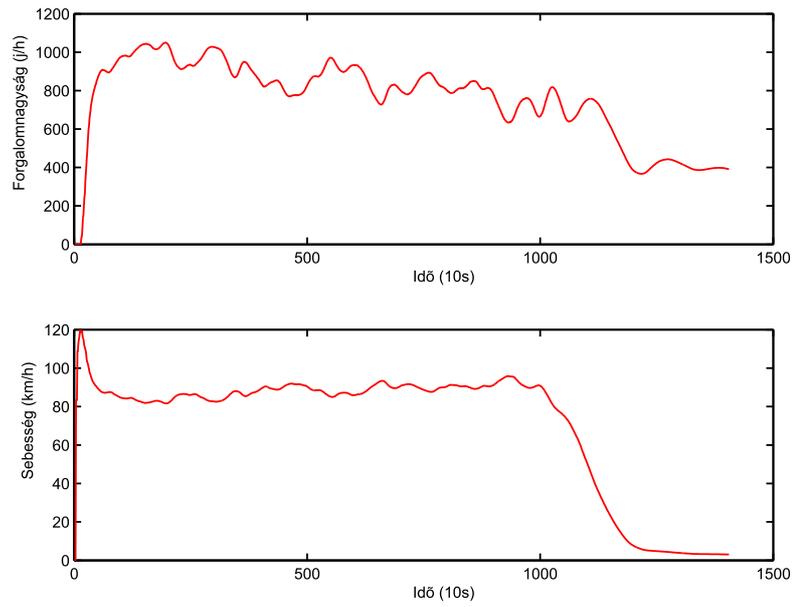
2.3.3. Modell értékelés

Az így beállított paraméterek mellett kapott modell és a valós értékek összehasonlítását a következő ábrák mutatják (késsel a valós adatok, pirossal pedig a METANET modell által számított adatok):



2.3. ábra. Modellértékek és valós adatok összehasonlítása

Mint az az ábrán jól látható a forgalomnagyság-görbe esetén a modell jól megközelíti a valós adatokat. Ugyanakkor a sebesség-görbe látszólag kevésbé. Ez a modell sebességre vonatkozó egyenletével magyarázható. A szegmensekben ugyanis átlagsebességeket számol a modell és azt "adja" tovább a következő szegmens számára a határfeltételeknek megfelelően, vagyis az összekapcsolás során a sebességértékek nagymértékben simulnak. Azonban - ami számunkra fontos - a forgalomban bekövetkező változásokat precízen követni képes. Erre vonatkozólag szimulációkat végeztem. Példaként a 7-es szegmensben balesetet szimuláltam, a kihaladó járműszám hirtelen lecsökkentésével. Ez esetben az 5. szegmens válaszát a következő ábra szemlélteti:



2.4. ábra. Modell válasza szimulált esemény esetén

Összefoglalva elmondhatom, hogy a felépített és kalibrált modell alkalmas a forgalom lefolyás szimulálására továbbá a forgalomban bekövetkező változások modellezésére és így az Automatikus Esemény Érzékeléshez alapul szolgálhat.

3. LINEÁRIS KALMAN-SZŰRŐ

3.1. Egyenletek levezetése

A Kalman-szűrő a rendelkezésre álló mérések alapján becsüli a rendszer állapotát. Tekintjük a következő zajokkal terhelt diszkrétidejű lineáris rendszert:

$$x_{k+1} = A_k x_k + B_k u_k + \Gamma w_k, \quad x(0) \quad (3.1)$$

$$y_k = C_k x_k + v_k. \quad (3.2)$$

Ahol x_k jelöli a rendszer állapotát a "k-adik" pillanatban, $x(0)$ kezdeti állapot, u_k a determinisztikus jel, w_k a folyamatzaj, v_k a mérési zaj, y_k a rendszer kimenete, továbbá Γ a folyamatzaj súlyozására szolgáló mátrix. A rendszert terhelő zajokra és a kezdeti állapotra a következő sztochasztikus hipotézis legyen érvényben:

$x(0)$ független w_k -től és v_k -től,

$$E x(0) = x_0, \quad E[(x(0) - x_0)(x(0) - x_0)^T] = \Sigma_0 \geq 0 \text{ (pozitív szemidefinit)}$$

$$E w_k = 0, \quad E[w_k w_l^T] = R_{w,k} \delta_{kl}, \quad R_{w,k} \geq 0 \text{ (pozitív szemidefinit)}$$

$$E v_k = 0, \quad E[v_k v_l^T] = R_{v,k} \delta_{kl}, \quad R_{v,k} > 0 \text{ (pozitív definit)}$$

$$E[w_k v_l^T] = 0, \quad E[v_l w_k^T] = 0, \text{ (} w_k \text{ és } v_k \text{ korrelálatlanok).}$$

Keressük tehát azt a lineáris szűrőt, amely az u_k és y_{k+1} mérési eredményekből az x_k állapot optimális \hat{x}_k becslését adja abban az értelemben, hogy:

$$E[x_k - \hat{x}_k] = 0 \quad \text{minden } k \text{ esetén,} \quad (3.3)$$

$$E \left[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^T \right] = \Sigma_k \rightarrow \text{infimum.} \quad (3.4)$$

Jelölje \hat{x}_k^- az ún. *a priori* állapotbecslésünk a k -edik lépésben, és \hat{x}_k az ún. *a posteriori* állapotbecslésünk a k -edik lépésben. Ezek után definiálhatjuk az *a priori* és az *a posteriori* becslési hibákat a következőképpen:

$$\begin{aligned} e_k^- &\equiv x_k - \hat{x}_k^-, \quad \text{és} \\ e_k &\equiv x_k - \hat{x}_k. \end{aligned}$$

Ekkor az *a priori* becslési hiba kovariancia:

$$P_k^- = E \left[e_k^- (e_k^-)^T \right],$$

illetve az *a posteriori* becslési hiba kovariancia:

$$P_k = E [e_k(e_k)^T].$$

Az *a posteriori* állapot becslés \hat{x}_k lineáris kombinációja az *a priori* becslés \hat{x}_k^- és az aktuális mérés z_k illetve a mérési előrejelzés $C\hat{x}_k^-$ súlyozott különbségének:

$$\hat{x}_k = \hat{x}_k^- + K_k(y_k - C\hat{x}_k^-) \quad (3.5)$$

Az egyenletben szereplő $(z_k - C\hat{x}_k^-)$ tagot nevezzük a mérési "maradéknak", ez a tag fejezi ki az eltérést az előrejelzett mérés $C\hat{x}_k^-$ és az aktuális mérés y_k között. Amennyiben a maradéktag értéke zérus, úgy a két érték teljesen megegyezik.

Az egyenletben szereplő $n \times m$ -es K mátrix szolgál az *a posteriori* becslés hiba-kovarianciájának minimalizálására. Számolások után K értékére a következő alak adódik [9]:

$$K_k = P_k^- C^T (C P_k^- C^T + V)^{-1} \quad (3.6)$$

Az egyenletet vizsgálva láthatjuk, hogy amennyiben V értéke, vagyis a mérési zaj kovarianciája nullába tart, akkor a K_k erősítés a maradéktagot erősebben súlyozza. Speciálisan:

$$\lim_{R_k \rightarrow 0} K_k = C^{-1}.$$

Viszont amennyiben az *a priori* becslési hiba kovarianciája, tehát P_k^- tart a nullába, úgy a K_k erősítés a maradéktagot kevésbé súlyozza, speciálisan [10]:

$$\lim_{P_k^- \rightarrow 0} K_k = 0.$$

3.1.1. A diszkrét lineáris Kalman-szűrő algoritmus

A Kalman szűrő tehát az állapotbecsléshez visszacsatolást használ a következőképpen: a szűrő becsli a folyamat állapotát egy időpontban, majd a (zajos) mérések által visszacsatolást alkalmaz. Ez alapján az algoritmus egyenleteit két csoportba oszthatjuk: *frissítés mérési időpontok közötti* egyenletek (time update between measurements), illetve *mérési eredmény frissítésére* vonatkozó egyenletek (measurement update). A mérési időpontok közötti egyenletek által történik az aktuális állapot és hiba kovariancia becslések időben előrejelzése és ezáltal az *a priori* becslések szolgáltatása a következő időpontra. A mérési eredmény frissítésére vonatkozó egyenletek pedig a visszacsatolást szolgáltatják, felhasználva az új mérési eredményt és az *a priori* becsléseket, egy módosított, jobb *a posteriori* becslést nyújtanak. Tehát a diszkrét lineáris Kalman-szűrő működését a következőképpen írhatjuk le:

A rendszer kezdeti állapotait, kezdeti becsléseket adunk meg, vagyis \hat{x}_0 -t és P_0 -t. Az algoritmus első szakasza a frissítés mérési időpontok között - vagy más néven "előrejelzés":

1. Az állapot előrejelzése:

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1}$$

2. A hiba kovariancia előrejelzése:

$$P_k^- = AP_{k-1}A^T + \Gamma W \Gamma^T$$

Az így kapott értékeket illetve a méréseket felhasználva következik a második szakasz a "korrekció":

1. Kalman-erősítés számolása:

$$K_k = P_k^- C^T (C P_k^- C^T + V)^{-1}$$

2. Állapotbecslés frissítése a mérési eredmények felhasználásával:

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - C\hat{x}_k^-)$$

3. Hiba kovariancia frissítése [9]:

$$P_k = (I - K_k C) P_k^-$$

3.2. A lineáris Kalman-szűrő hangolása

3.2.1. A Kalman-szűrő frekvencia-tartománybeli tulajdonsága

A megfelelő eredmény elérése érdekében a lineáris Kalman-szűrőt hangolni szükséges. Tekintsük ismét a következő diszkrét idejű, zajjal terhelt rendszert:

$$x_{k+1} = A_k x_k + B_k u_k + \Gamma w_k, \quad x(0) \quad (3.7)$$

$$y_k = C_k x_k + v_k. \quad (3.8)$$

Mely egyenletekből a már bemutatott K Kalman-szűrő erősítésre a következő összefüggés adódik:

$$K = \Sigma C^T V^{-1} \quad (3.9)$$

ahol Σ jelöli az állapotbecslés hiba-mátrixát, mely megoldása a következő ún. "Filter Algebrai Ricatti Egyenletnek" (FARE):

$$\Sigma A^T + A \Sigma - \Sigma C^T V^{-1} C \Sigma + \Gamma W \Gamma^T = 0. \quad (3.10)$$

Itt jegyezném meg, hogy a szűrési probléma eredményeit az LQ irányítás duálisaként kaphatjuk meg. LQ optimális irányítás esetén az optimális állapot-visszacsatolás mátrixa a következő alakban állt elő:

$$K = R^{-1} B^T P,$$

ahol P megoldása a következő - híres - algebrai Ricatti egyenletnek:

$$PA + A^T P - PBR^{-1}B^T P + Q = 0.$$

Az egyenletben szereplő Q illetve R mátrixok mint ismeretes az állapotok illetve bemenetek súlyozására szolgál az LQ irányítás kvadratikus funkcionál egyenletében. Amennyiben a CARE egyenletben a következő "helyettesítéseket" végezzük el, vagyis a duálisát képezzük:

$$A \rightarrow A^T; B \rightarrow C^T; R \rightarrow V; Q \rightarrow \Gamma W \Gamma^T$$

úgy a már ismertetett FARE egyenlethez jutunk.

Tegyük fel, az egyszerűség kedvéért hogy a vizsgált rendszerünk esetén nincs bemeneti jel, vagyis $u \equiv 0$. Ezt megtehetjük, hiszen mint láttuk a bemenőjel azonos mértékben hat x -re és \hat{x} -ra egyaránt. A Kalman-szűrő K mátrixának bemenetére az átviteli függvény a következő alakban adódik:

$$G_{KF}(s) = C\phi(s)K, \quad (3.11)$$

ahol $\phi(s) = (sI - A)^{-1}$, továbbá az eltérés, illetve hiba definícióját felhasználva:

$$E(s) = Y(s) - \hat{Y}(s) = Y(s) - C\phi(s)KE(s), \quad (3.12)$$

amiből adódik:

$$E(s) = [I + C\phi(s)K]^{-1} Y(s). \quad (3.13)$$

Az egyenletben megjelenő $[I + C\phi(s)K]^{-1}$ nem más, mint a hurok érzékenységi függvénye, melyet a továbbiakban S_{KF} -el jelölhetünk. Definíció szerint a hurok kiegészítő érzékenységi függvénye: $T_{KF}(s) = I - S_{KF}(s)$, vagyis:

$$T_{KF}(s) = I - S_{KF}(s) = I - [I + G_{KF}(s)]^{-1} = G_{KF}(s)[I + G_{KF}(s)]^{-1}. \quad (3.14)$$

A FARE egyenletből levezethető a következő összefüggés, mely az ún. Kalman Frequency Domain Identity, röviden KFDI¹:

$$[I + G_{KF}(s)]V[I + G_{KF}(-s)]^T = V + [C\phi(s)\Gamma]W[C\phi(-s)\Gamma]^T. \quad (3.15)$$

Mielőtt a Kalman-szűrő hangolására rátérnénk, először az ehhez szükséges szinguláris értékekről lesz szó.[10]

3.2.2. Szinguláris értékek

Egyváltozós SISO esetben a visszacsatolt ág teljesítményének vizsgálata a felnyitott hurok erősítés karakterisztikája adott választ (Bode, Nyquist). Amennyiben az egyváltozós esetben érvényes összefüggéseket szeretnénk többváltozós MIMO esetre alkalmazni a probléma a következő: a $\|G(s)u(s)\|$ norma értéke függ az $u(s)$ vektor irányától. Azonban korlátozhatjuk ún. mátrix norma használatával:

$$\frac{\|G(s)u(s)\|}{\|u(s)\|}.$$

¹ Az LQ irányítással kapcsolatos dualitás természetesen itt is érvényes. A KFDI LQ esetben a következő alakú: $[I + G_{LQ}(-s)]^T R [I + G_{LQ}(s)] = R + [N\phi(-s)B]^T [N\phi(s)B]$ ahol $Q = N^T N$.

Vagyis egyetlen erősítés érték helyett egy tartományt kaptunk, mely alulról és felülről korlátozva van.

Amennyiben $\|x\|$ jelöli x vektor bármilyen normáját, úgy az ún. indukált mátrix normát a következőképpen definiáljuk:

$$\|G\| = \sup_{x \neq 0} \frac{\|Gx\|}{\|x\|}.$$

Speciálisan: ha Euklideszi normát használunk, vagyis

$$\|x\| = \sqrt{(x^*x)^2}$$

akkor az indukált mátrix norma az ún. Hilbert norma:

$$\|G\|_s = \bar{\sigma},$$

ahol $\bar{\sigma}^2$ a G^*G (vagy GG^*) maximális sajátértéke. A G^*G (illetve GG^*) sajátértékeinek pozitív négyzetgyökeit nevezzük G szinguláris értékeinek.

Amennyiben G helyett $G(s)$ -el számolunk, továbbá $s = i\omega$, úgy $G(i\omega)$ szinguláris értékei ω függvényei, és ezeket a nevezhetjük a $G(s)$ átviteli függvény erősítéseinek [7].

3.2.3. Hangolási lehetőségek

Tegyük fel, hogy az egyenletben szereplő várható értékek a következő alakúak: $V = \mu I_n$ illetve $W = I_n$. Ekkor a KFDI:

$$[I + G_{KF}(s)][I + G_{KF}(-s)]^T = I + \frac{1}{\mu} [C\phi(s)\Gamma][C\phi(-s)\Gamma]^T \quad (3.16)$$

Az $s = i\omega$ helyettesítés után, az KFDI szinguláris érték alakja:

$$\sigma_i \{I + G_{KF}(i\omega)\} = \sqrt{1 + \frac{1}{\mu} \sigma_i^2 \{C\phi(i\omega)\Gamma\}} \quad (3.17)$$

² x^* jelöli a vektor komplex-konjugáltját, vagyis \bar{x}^T -el egyenértékű.

Vagyis a következő eredményre jutottunk:

- μ értékével szabályozható a Kalman-szűrő "hurok-erősítése" (a szinguláris értéket "súlyozza")
- Γ határozza meg a $C\phi(i\omega)\Gamma$ zérusait és ezáltal a szinguláris érték alakját a frekvencia tartományban

A fejezet [5], [6], [7], [9], [10] és [12] felhasználásával készült.

3.3. Lineáris Kalman-szűrő felhasználása forgalmi változók becslésére

3.3.1. A modell linearizálása

Amint azt láttuk a lineáris Kalman-szűrő aszimptotikusan stabil állapotbecslő. Ez alapján felhasználhatjuk a vizsgált forgalmi változók értékeinek becslésére.

Módosítsuk a METANET-modellt valós esetre, vagyis terhelje a folyamatot illetve a megfigyelést zaj. Ekkor a modell egyenletek a következő alakban írhatóak:

$$\begin{aligned}\rho_i(k+1) &= \rho_i(k) + \frac{T}{\Delta_i n} [q_{i-1}(k) - q_i(k) + r_i(k) - s_i(k)] \\ v_i(k+1) &= v_i(k) \\ &+ \frac{T}{\tau} [V(\rho_i(k)) - v_i(k)] \\ &+ \frac{T}{\Delta_i} v_i(k) [v_{i-1}(k) - v_i(k)]\end{aligned}\tag{3.18}$$

$$\begin{aligned}&- \frac{\nu T}{\tau \Delta_i} \frac{\rho_{i+1} - \rho_i(k)}{\rho_i(k) + \kappa} \\ &+ \xi_k^v \\ q_i(k) &= \rho_i(k) \cdot v_i(k) \cdot n(k) + \xi_k^q\end{aligned}\tag{3.19}$$

Az 3.18 egyenletben megjelenő ξ_k^v a sebességre jellemző folyamatzaj a k -edik lépésben, míg hasonlóan az 3.18 egyenletben ξ_k^q a forgalomnagyságra jellemző folyamatzaj a k -edik lépésben. Mint látható a forgalomsűrűsége vonatkozó egyenletet nem terheli folyamatzaj, hiszen - ha még emlékszünk - ez a forgalom megmaradás alapegyenlete, amelyik minden időpillanatban igaz. A ξ^v és ξ^q zajokat 3.1-nak megfelelően zérus várható értékű Gauss normál eloszlású fehér zajnak tekintjük.

Ahhoz, hogy a lineáris Kalman-szűrőt alkalmazni tudjam a forgalmi modellre, linearizálni kellett a nem-lináris modellegyenleteket. A linearizálással előálló állapotter-reprezentációra terveztem a Kalman-szűrőt. Bemenő jelnek a szakaszba belépő forgalomnagyságot választottam, míg kimenő jelnek az első szegmensbe belépő járművek sebességét, a szakaszból kilépő forgalomnagyságot és a kilépő járművek átlagsebességét. Hogy miért így választottam meg, azt a linearizálás elvégzése után válaszolom meg.

A linearizálás során a következő módon jártam el. Először az egyes szegmenseket linearizáltam egy általam választott munkapont körül. Ezt a munkapontot úgy választottam meg,

hogy a forgalom normál lefolyását jellemezze. A vizsgált időtartam során az átlagos forgalom-nagyság, forgalomsűrűség illetve átlagsebesség jelentette a munkapontot, mely a szegmensek esetében mint állapot és mint bemenőjel is megjelent. A szegmenseket ugyanis, mint láthattuk egymáshoz kapcsolva kapjuk az autópálya szakasz modellét. Továbbá a szegmensek linearizálása során figyelembe vettem a felhajtó-ág forgalmát és a szegmensek hosszát is. Ennek megfelelően három munkapont körül linearizáltam a szegmenseket:

1. $\rho_{munkapont}, v_{munkapont}, q_{munkapont}, \Delta_1$ felhajtó forgalom nulla,
2. $\rho_{munkapont}, v_{munkapont}, q_{munkapont}, \Delta_1$ felhajtó forgalom 50,
3. $\rho_{munkapont}, v_{munkapont}, q_{munkapont}, \Delta_2$ felhajtó forgalom nulla.

A linearizálást elvégezve rendelkezéseimre állt az egyes szegmensek (A_i, B_i, C_i, D_i) állapotter reprezentációja. Ezen linearizált szegmenseket kapcsoltam ezután össze, mely eredményeként a szakaszra jellemző $(A_{MN}, B_{MN}, C_{MN}, D_{MN})$ állapotter reprezentáció állt elő, ahol:

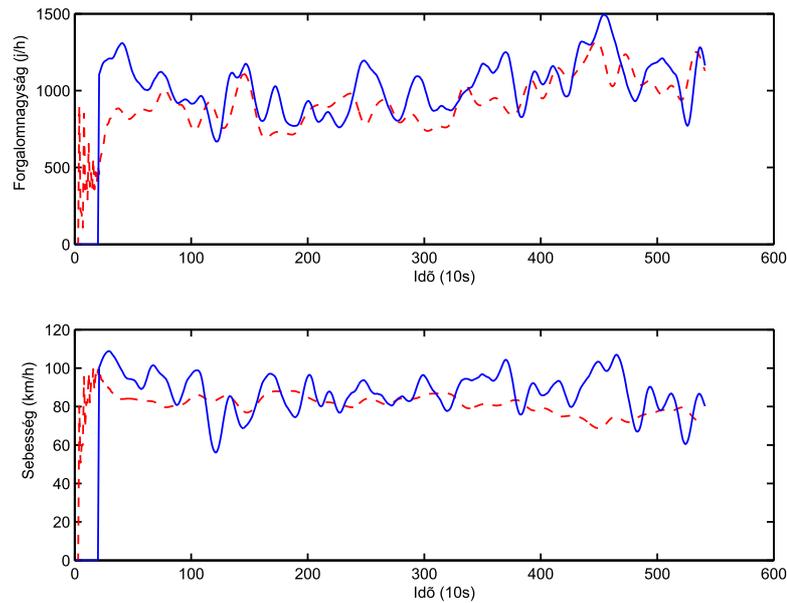
A_{MN} egy 24×24 -es mátrix, mely a nyolc szegmens három-három állapotváltozója között fejezi ki a kapcsolatot

B_{MN} egy 24×5 -es mátrix, mely a bemenet q_{belp} és felhajtó forgalmak, szegmenshosszak mint bemenetek illetve az állapotváltozók közötti kapcsolatot reprezentálja

C_{MN} pedig 3×24 -es mátrix, hiszen a kimenetek v_{kilp}, v_{kilp} és q_{kilp} jellemzi

D_{MN} 3×5 nullmátrix, a belépő és a kilépő változók között közvetlen hatás nincs

A linearizált modell és a valós adatok összehasonlítását az ellenőrzőponton a következő ábra mutatja (pirossal a linearizált modell, kék színnel pedig a valós adatok):



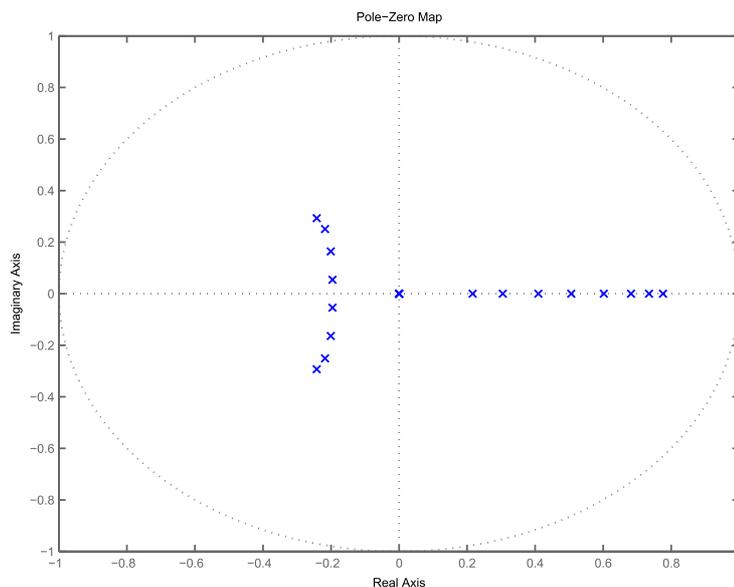
3.1. ábra. Lineáris modell összehasonlítása a valós adatokkal

Az ábrán jól látható, hogy forgalomnagyság esetén a linearizált modell jól közelíti a valós adatokat, míg a sebesség görbe esetén a már ismertetett probléma áll fenn továbbra is.

A linearizálással előállt a vizsgált szakasz állapotter reprezentációja, kérdéses azonban, hogy a rendszerre alkalmazhatjuk-e a Kalman-szűrő algoritmusát. Ehhez először a állapotter megfigyelhetőségi mátrixát vizsgáltam. Ennek eredménye:

$$\text{rank}(\mathcal{O}) = 21 \neq n = 24,$$

vagyis a rendszer nem teljesen megfigyelhető. A Kalman-szűrő alkalmazásának feltétele, hogy a rendszer *detektálható* legyen. Egy rendszert akkor nevezünk detektálhatónak, ha a nem megfigyelhető sajátértékek stabilak. Ehhez a rendszer pólusait vizsgáltam meg, melyet a következő ábra szemléltet:



3.2. ábra. A linearizált METANET modell pólusai

A diszkrét rendszerekre vonatkozó Nyquist stabilitási kritérium értelmében a rendszer minden pólusa (így a rendszer is) stabil, hiszen az origó középpontú egységsugarú körön belül találhatóak. Tehát a lineáris Kalman-szűrő technikát alkalmazhatjuk a METANET-modell esetében a forgalmi változók becslésére, hiszen a linearizált rendszer sajátértékei stabilak.

Ezek után már megválaszolhatjuk az input/output megválasztására vonatkozó kérdést. Azzal, hogy a szakasz elején belépő forgalom sebességét is kimenetnek tekintjük a rendszer megfigyelhetősége - és ezáltal a Kalman-szűrés minősége - is javul. Ahhoz, hogy az első szegmensbe érkező járművek sebessége kimenet legyen, egyszerűen az első szegmens végén számított értékekkel helyettesítettem.

3.3.2. Modellelés

A SIMULINK környezetben felépített rendszer modellje a B.3 látható. Azzal a feltételezéssel éltem, hogy a beolvasott adatok (simított értékek) a forgalom valós adatai. Tehát ezek tartalmazzák az 3.18, 3.18 egyenletekben feltüntetett folyamat zajokat. Azt is láttuk, hogy a forgalomsűrűség egyenletét nem terheli folyamatzaj. Ezek után a mérési egyenleteik módosítsuk. Az elhelyezett detektorok által mért értékek q_m illetve v_m eltérnek a valós q_v és v_v értékektől. Vagyis:

$$q_m = q_v + \eta_q \quad (3.20)$$

$$v_m = v_v + \eta_v \quad (3.21)$$

ahol η_q , η_v jellemzi a mérési bizonytalanságot. Ezen zajokra a már ismert zérus várható értékű Gauss normál eloszlású hipotézis áll fenn. A modell felépítésén látszik, hogy ezeket a szenzorajokat a simított adatokra tettem rá.

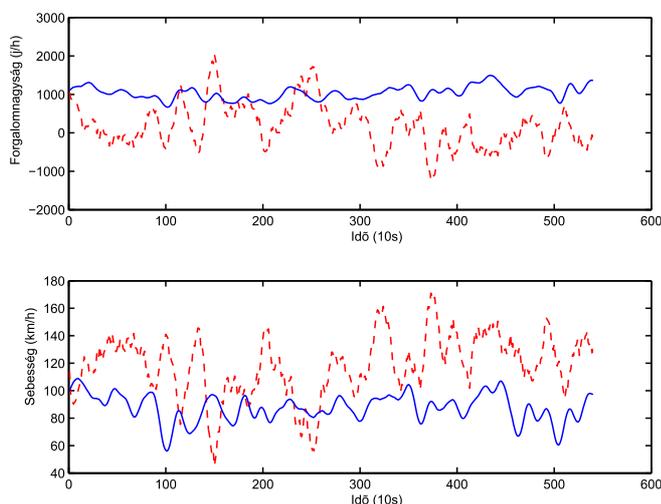
A modell felépítésekor, Kalman-szűrő alkalmazásához szükséges megadni a kezdeti állapot illetve hiba kovariancia értékeket. Az állapotok esetén a már ismert "normál" forgalomra

vonatkozó értékeket állítottam be, míg a hiba kovariancia mátrix értékeit egységnyinek választottam. Amennyiben a hiba kovariancia mátrix értékét zérusnak választanánk az a Kalman-szűrő egyenleteit tekintve azt jelentené, hogy a kezdeti értékeink pontosak, vagyis a szűrés nem valósulna meg³.

A modellben még egy érdekes pont van, mint az látható a Kalman-szűrő bemenete előtt és után a munkapont értékével csökkenteni illetve növelni kell a belépő/kilépő jelet. Erre azért van szükség, mert a linearizált rendszer esetén a munkapont körüli ingadozást vizsgáljuk. Ezeket a megfontolások figyelembe vétele után hangolhatjuk be a lineáris Kalman-szűrőt.

3.3.3. A lineáris Kalman-szűrő hangolása, frekvenciatartománybeli analízis

A rendszer frekvenciatartománybeli analízisének a levezetésben már ismerttetett alapokból indultam ki. W illetve V értékeit egységmátrixnak feltételeztem, kiindulásként pedig Γ illetve μ értékeit is egységnyire választottam. A rendszer állandósult állapotbeli válaszát hasonlítsuk össze a valós adatokkal (pirossal az állapotbecslés, kézzel a valós adatok):



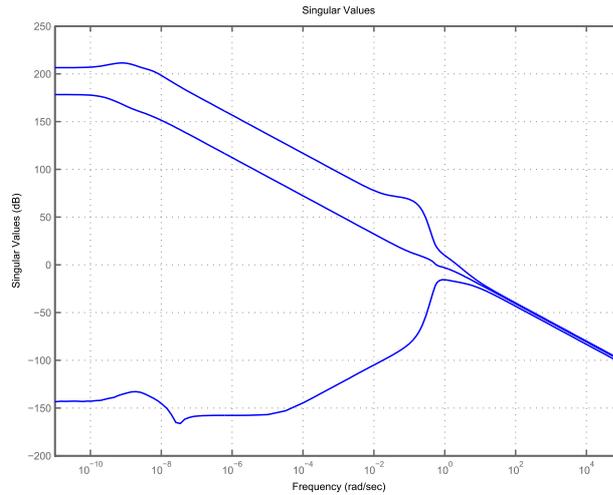
3.3. ábra. A becslt és a valós adatok az ellenőrzőponton

Mint látható az állapotbecslés nem megfelelő, ezért módosításra van szükség. A frekvencia-tartománybeli analízis során a felnyitott hurok vagyis:

$$c\phi(z)K_f$$

szinguláris értékeit vizsgáltam. A rendszer állandósult állapotbeli szinguláris érték görbéje:

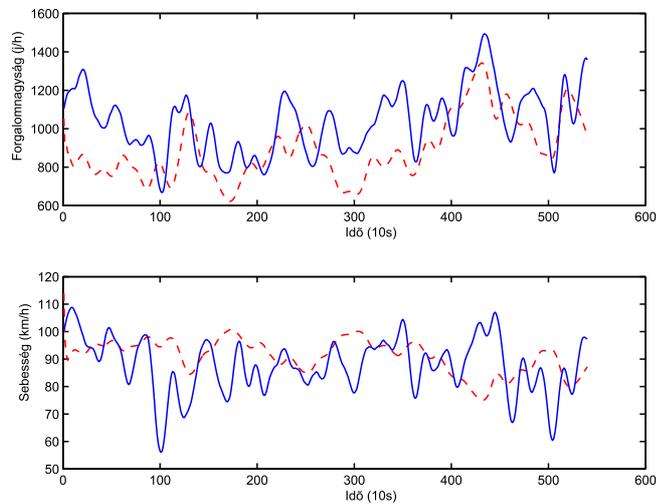
³ Ugyanis, emlékeztetőül: $\lim_{P_k^- \rightarrow 0} K_k = 0$



3.4. ábra. A felnyitott hurok szinguláris értékei

A vágási körfrekvencia: $\omega_c \approx 1 \frac{rad}{sec}$.

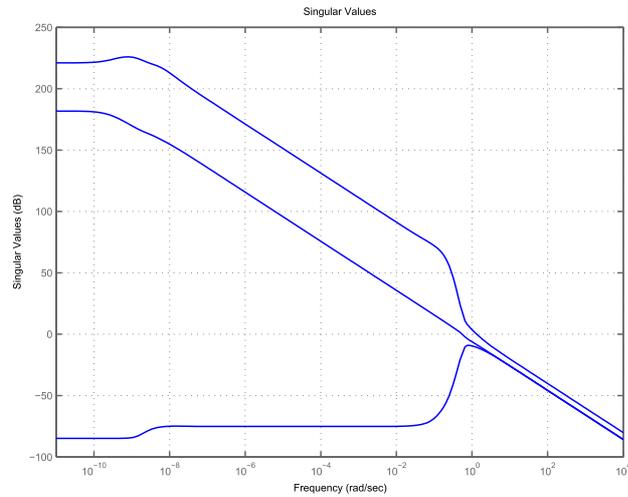
Második lépésben a METANET modell egyenletek alapján a forgalomsűrűség egyenletét terhelő zajt zérusra változtattam, Γ megfelelő értékeinek zérusra történő módosításával. A rendszer időtartománybeli válasza az ellenőrzőponton (pirossal a becült, kézzel a valós adatok):



3.5. ábra. A becült és a valós adatok az ellenőrzőponton

A forgalomsűrűségekre vonatkozó zaj nullára csökkentésével mint láthatjuk jobb állapotbecslést értünk el. A becült és a valós adatok már közel azonos értéktartományban mozognak, mind a sebesség, mind a forgalomnagyság esetében, ám az állapotbecslés még nem megfelelő.

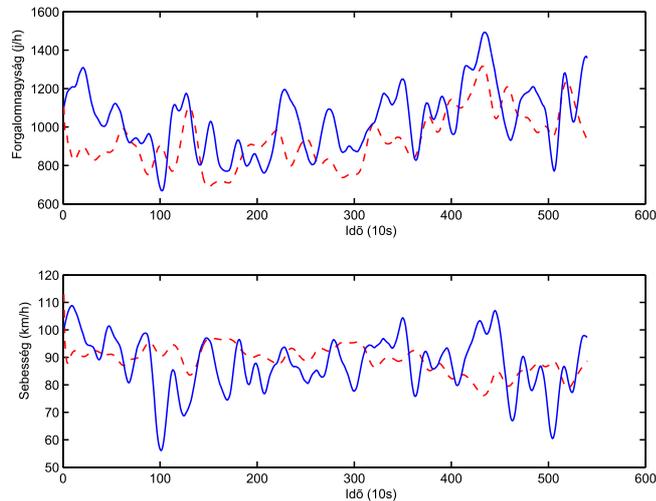
A felnyitott hurok állandósult állapotbeli szinguláris érték görbéje:



3.6. ábra. A rendszer szinguláris értékei

A vágási körfrekvencia értéke $\omega_c \approx 1 \frac{rad}{sec}$. Látható, hogy alacsony frekvencián az erősítést stabilizáltuk. Illetve Γ módosításával a szinguláris görbe alakja megváltozott.

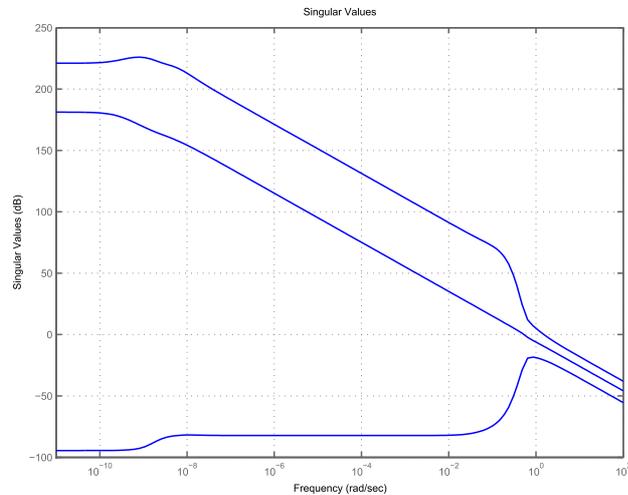
Harmadik lépésben a mérési zaj kovarianciájának súlyozását a következőképpen módosítottam. A mérési zajokat én tettem a modellre, mely során azzal a feltételezéssel éltem, hogy a forgalomnagyság esetén a szenzor zaj nagyobb mint a sebesség esetén. Ezen módosításokat a μ értékeinél is figyelembe vettem. Ekkor az ellenőrzőponton (ismét pirossal az állapotbecslés és kék színnel a valós értékek):



3.7. ábra. A becsült és a valós adatok az ellenőrzőponton

Mint látjuk az állapotbecslés tovább javult.

A frekvencia tartományban a szinguláris érték görbe:



3.8. ábra. A rendszer szinguláris értékei

A vágási körfrekvencia $\omega_c \approx 1 \frac{rad}{sec}$.

Eddigi eredményeink a következőket mutatják: az állapotbecslés időtartományban megfelelő. Frekvencia tartományban vizsgálva:

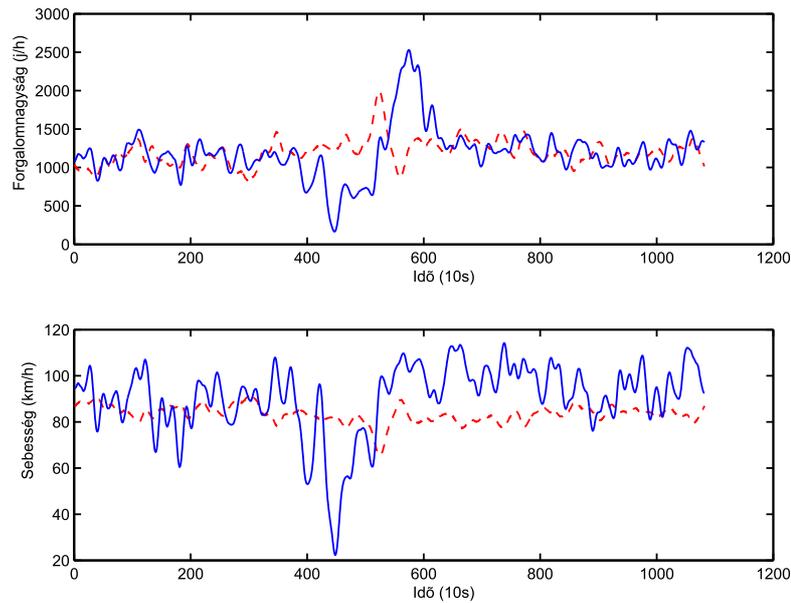
- Alacsony szenzor zaj esetén ($\mu \approx 0$) a szűrő átenged minden frekvenciát, vagyis nem szűr.
- Alacsony folyamat zaj esetén ($\Gamma \approx 0$) a szűrő annyira szűr amennyira csak képes, vagyis az állapotbecslés romlik.

A rendszer a vágási körfrekvencia körül jól kondicionált, vagyis a maximális és minimális szinguláris értékek aránya egy:

$$\frac{\bar{\sigma}}{\underline{\sigma}} \approx 1$$

3.3.4. Lineáris Kalman-szűrő baleseti körülmények mellett

A vizsgált napon történt egy baleset. Nézzük hogyan "reagál" a szűrő a dinamika változására, az ellenőrzőponton a becsült és valós forgalmi változók (pirossal az állapotbecslés, kézzel a valós adatok):



3.9. ábra. Lineáris állapotbecslés és valós adatok összehasonlítása az ellenőrzőponton

A lineáris Kalman-szűrő baleset esetén nem tudja megfelelő mértékben követni a dinamika változását. Ennek magyarázata két okra vezethető vissza. Egyrészt baleset esetén a forgalmi változók értékei elhagyják a lineáris tartományt és így a felállított állapotter-reprezentáció nem érvényes. Másrészt az 2.3 egyenletünk csak homogén forgalom esetén igaz, inhomogén forgalom - baleseti körülmények - esetén már nem. Ezeknek a problémáknak megoldásáról - a megoldás közelítéséről - a következő, Kiterjesztett Kalman-szűrőről szóló fejezetben lesz szó.

Összefoglalva elmondhatom, hogy automatikus eseményérzékeléshez lineáris Kalman-szűrő nem alkalmazható, hiszen a modell nem-lineáris. A linearizálási tartományon belül az állapotbecslés megfelelő, azonban a tartományból kilépve (baleset) a minőség romlik, az állapotbecslés nem képes megfelelő mértékben követni a változásokat. Azonban nem volt eredménytelen a fejezet, hiszen a Kalman-szűrő hangolásáról kaptunk információkat, melyeket a kiterjesztett esetben hasznosítani tudunk majd.

4. KITERJESZTETT KALMAN-SZŰRŐ

A fejezet a nem-lineáris rendszerek esetében használható kiterjesztett Kalman-szűrőről és annak felhasználásáról szól. Az előző - lineáris Kalman-szűrőről - szóló fejezet felépítését követve az első rész az elméleti alapokról szól. Hasonlóan a lineáris Kalman-szűrőnél az egyenletek levezetésével betekintést nyerhetünk a nem-lineáris állapotbecslés megoldásába.

A második részben pedig az elméleti alapok gyakorlati megvalósításáról lesz szó. Bemutatom, hogy miként lehet alkalmazni a kiterjesztett Kalman-szűrő algoritmusát a nem-lineáris METANET-modellben. Az állapotbecslés meg nem oldott problémájára is közelítő megoldást mutatok be, mellyel lehetőség nyílik az AID rendszer felépítésére.

Végül a kiterjesztett Kalman-szűrési technikát felhasználva a modell-paraméterek "on-line" kalibrációjának gondolatmenetét mutatom be. Ezt a technikát két fontos paraméter esetén alkalmaztam, mely segítségével a forgalom lefolyásáról plusz információkat is kaphatunk.

4.1. Egyenletek levezetése

Az előző fejezetben bemutatott Kalman-szűrő lineáris rendszerek esetében használható. Azonban mint láttuk a használni kívánt modell nem-lineáris, így az állapotbecslés romlik a lineáris tartományon kívül. Nem-lineáris esetben a Kalman-szűrő az aktuális várható érték és kovariancia felhasználásával linearizál, ezt nevezzük *Kiterjesztett Kalman szűrőnek* vagy angolul *Extended Kalman Filter - EKF*.

Tehát a linearizálást az aktuális állapotbecslés körül végezzük, felhasználva a folyamatot leíró függvények parciális deriváltjait. Legyen tehát a folyamat a következő:

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}), \quad (4.1)$$

illetve a mérési egyenlet:

$$y_k = h(x_k, v_k). \quad (4.2)$$

Az egyenletekben szereplő w_k és v_k ismét a folyamatra illetve mérésre jellemző bizonytalanságot reprezentálja, az f illetve h pedig a nem-lineáris vektorfüggvényeket jelentenek.

A rendszert terhelő zajok értékeit nem tudjuk minden lépésben. Ezért közelíthetjük az állapot- illetve mérési vektorokat zérus értékű zaj mellett:

$$\tilde{x}_k = f(\hat{x}_{k-1}, u_{k-1}, 0) \quad (4.3)$$

$$\tilde{y}_k = h(\tilde{x}_k, 0), \quad (4.4)$$

ahol \hat{x}_k jelöli az állapot *a posteriori* becslését az előző időpillanat alapján. A nem-lineáris függvényeket e becslések körül fogjuk linearizálni, vagyis:

$$x_k \approx \tilde{x}_k + A(x_{k-1} - \hat{x}_{k-1}) + \Gamma w_{k-1}, \quad (4.5)$$

$$z_k \approx \tilde{z}_k + C(x_k - \tilde{x}_k) + \Lambda v_k. \quad (4.6)$$

Ahol

- x_k és y_k az aktuális állapot és mérési vektorok,
- \tilde{x}_k és \tilde{z}_k az egyenletekben szereplő közelítő állapot és mérési vektorok,
- \hat{x}_k az *a posteriori* állapotbecslés a k -edik lépésben,
- A az f x -szerinti parciális deriváltjait tartalmazó Jacobi mátrix:

$$A_{ij} = \frac{\partial f_i}{\partial x_j}(\hat{x}_{k-1}, u_{k-1}, 0),$$

- Γ az f w -szerinti parciális deriváltjait tartalmazó Jacobi mátrix:

$$\Gamma_{ij} = \frac{\partial f_i}{\partial w_j}(\hat{x}_{k-1}, u_{k-1}, 0),$$

- C a h x -szerinti parciális deriváltjait tartalmazó Jacobi mátrix:

$$C_{ij} = \frac{\partial h_i}{\partial x_j}(\tilde{x}_{k-1}, 0),$$

- Λ a h v -szerinti parciális deriváltjait tartalmazó Jacobi mátrix:

$$\Lambda_{ij} = \frac{\partial h_i}{\partial v_j}(\tilde{x}_{k-1}, 0).$$

Az előrejelzés hibáját a következőképpen definiáljuk:

$$\tilde{e}_{x_k} \equiv x_k - \tilde{x}_k, \quad (4.7)$$

illetve a mérési maradékot:

$$\tilde{e}_{y_k} \equiv y_k - \tilde{y}_k. \quad (4.8)$$

Az egyenleteket felhasználva kapjuk a hiba folyamatra vonatkozó egyenleteinket:

$$\tilde{e}_{x_k} \approx A(x_{k-1} - \hat{x}_{k-1}) + \varepsilon_k, \quad (4.9)$$

$$\tilde{e}_{y_k} \approx C\tilde{e}_{x_k} + \eta_k, \quad (4.10)$$

ahol ε_k és η_k független sztochasztikus változók, melyek várható értékük zérus, kovarianciájuk pedig: $\Gamma W \Gamma^T$ illetve $\Lambda V \Lambda^T$.

A kapott egyenletek lineárisak és alakjuk 3.1-3.2-hoz hasonló. Ez alapján - az előző fejezet eredményei szerint - \tilde{e}_{y_k} mérési maradéktagot használhatjuk az \tilde{e}_{x_k} előrejelzési hiba becslésére és egy feltételes Kalman-szűrőt alkalmazunk. Ezt a becslést jelöljük \hat{e}_k és felhasználhatjuk az eredeti folyamat *a posteriori* állapotbecsléséhez:

$$\hat{x}_k = \tilde{x}_k + \hat{e}_k. \quad (4.11)$$

Az új véletlen változók eloszlásáról közelítőleg a következőket mondhatjuk:

$$\begin{aligned} p(\tilde{e}_{x_k}) &\sim N(0, E[\tilde{e}_{x_k} \tilde{e}_{x_k}^T]) \\ p(\varepsilon_k) &\sim N(0, \Gamma W \Gamma^T) \\ p(\eta_k) &\sim N(0, \Lambda V \Lambda^T) \end{aligned}$$

Ezeket a közelítéseket felhasználva az \hat{e}_k becslésére szolgáló feltételes Kalman szűrő egyenlete:

$$\hat{e}_k = K_k \tilde{e}_{y_k}. \quad (4.12)$$

Mely egyenletet felhasználva, kapjuk:

$$\begin{aligned} \hat{x}_k &= \tilde{x}_k + K_k \tilde{e}_{y_k} \\ &= \tilde{x}_k + K_k (y_k - \tilde{y}_k) \end{aligned} \quad (4.13)$$

Az egyenletet vizsgálva látjuk, hogy valójában a feltételes Kalman-szűrőre nincs szükségünk. Vagyis a kapott egyenletet használhatjuk a kiterjesztett Kalman-szűrő algoritmusában a mérési frissítésre [9].

4.1.1. Kiterjesztett Kalman-szűrő algoritmusa

Hasonlóan a lineáris esetben, az algoritmus a következő. A rendszer kezdeti állapotait, kezdeti becsléseket adunk meg, vagyis \hat{x}_0 -t és P_0 -t. Az algoritmus első szakasza a frissítés mérési időpontok között - vagy más néven "előrejelzés":

1. Az állapot előrejelzése:

$$\hat{x}_k^- = f(x_{k-1}, \hat{u}_{k-1}, 0)$$

2. A hiba kovariancia előrejelzése:

$$P_k^- = A_k P_{k-1} A_k^T + \Gamma_k W_k \Gamma_k^T$$

Az így kapott értékeket illetve a méréseket felhasználva következik a második szakasz a "korrekció":

1. Kalman-erősítés számolása:

$$K_k = P_k^- C_k^T (C_k P_k^- C_k^T + \Lambda_k V_k \Gamma_k^T)^{-1}$$

2. Állapotbecslés frissítése a mérési eredmények felhasználásával:

$$\hat{x}_k = \hat{x}_k^- + K_k (y_k - h(\hat{x}_k^-))$$

3. Hiba kovariancia frissítése [9]:

$$P_k = (I - K_k C_k) P_k^-$$

4.2. Kiterjesztett Kalman-szűrő alkalmazása forgalmi változók becslésére

Az előző szakasz végén bemutatott lineáris Kalman-szűrő eredményeit tekintve illetve a rendelkezésre álló kiterjesztett Kalman-szűrő technikát figyelembe véve kézenfekvőnek tűnik ezt az algoritmust alkalmazni a nem-lineáris egyenleteinkre.

Ehhez a következő B.4 nevű modellt építettem fel SIMULINK-ben. A rendszer alapja a pirossal jelzett A.6 *MATLAB* fájl, mely működését tekintve a kiterjesztett Kalman-szűrést valósítja meg.

A nem-lineáris Kalman-szűrő esetén hasonlóan jártam el, mint lineáris esetben. Bemenetnek ismét - a már vázolt megfontolások miatt - az első szegmens belépő forgalomnagyságát (q_0) választottam. A rendszer kimenete pedig ezúttal is az első szegmens belépő forgalmának térbeli átlagsebességét (v_0), illetve az utolsó szegmens kilépő forgalomnagyságát (q_8) és átlagsebessége (v_8).

A függvény bemenő jelként kapja meg a következő változók aktuális értékeit:

$u(k)$: bemenő jel, a szakaszba belépő forgalomnagysága a k -edik lépésben

$y(k)$: a szakasz "kimenő jele" a k -edik lépésben,

$$y_k = \begin{bmatrix} v_0 \\ v_8 \\ q_8 \end{bmatrix}$$

$x_{po}(k-1)$: az előző lépés állapotbecslése,

$P_{po}(k-1)$: az előző lépésben meghatározott becslési hiba kovariancia.

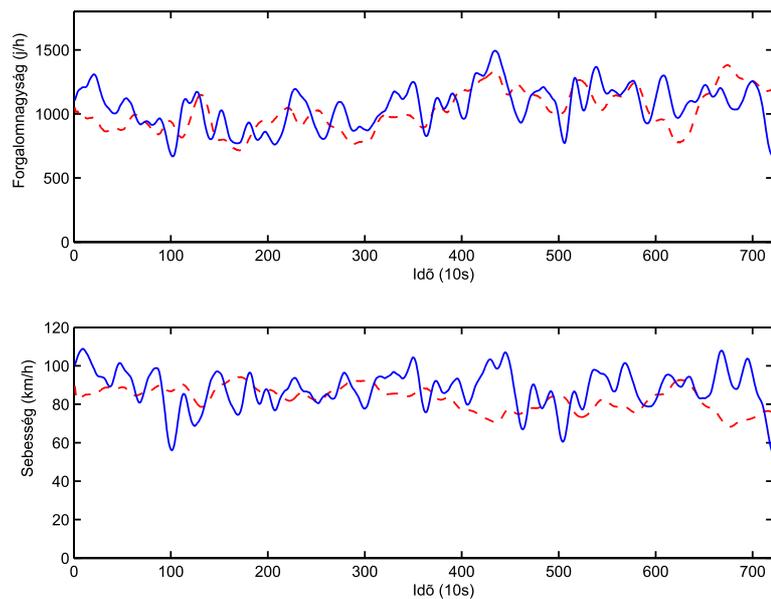
A felépített program tehát munkapontonként linearizálja a már bemutatott B.2 SIMULINK modellt az aktuális állapotbecslés körül. A visszacsatolt becslült állapotváltozókat először linearizálásra használjuk fel. Ez alapján előáll az egyes szegmensek adott munkapont körül érvényes A_i, B_i, C_i, D_i állapotter reprezentációja. Ezután a linearizált szegmenseket összekapcsoljuk, így kapjuk a linearizált $A_{MN}, B_{MN}, C_{MN}, D_{MN}$ szakaszra és aktuális szakasz-állapotra jellemző állapotter-reprezentáció. A kiterjesztett Kalman-szűrés során ezt az állapotter-reprezentációt használjuk fel. Az -egyenleteknek megfelelően az aktuális - visszacsatolt - állapotváltozók értékeivel kiszámítjuk a modell választát, ez képi majd az adott lépésben az *a priori*. Innentől az algoritmus a fent vázolt módon működik.

A kiterjesztett Kalman-szűrő esetén használatos Γ mátrix megegyezik a lineáris esetben használtal, hiszen a forgalom egyenleteit tartalmazó folyamatzaj és az állapot között lineáris függvénykapcsolatot tételeztünk fel. A Λ mátrix hasonlóan megfeleltethető a lineáris esetben használt μ -vel.

Kezdeti értéknek a lineáris szűrő esetén is használt értékeket állítottam be. A nulladik időpontban az állapotok értéke a normál forgalomlefolrás melletti értékek, míg a becslési hiba-kovariancia értéke minden állapotváltozó esetében egységnyi.

A rendszert terhelő zajok modellezésére a lineáris Kalman-szűrő frekvenciatartománybeli analízise során megállapított értékeket használtam fel.

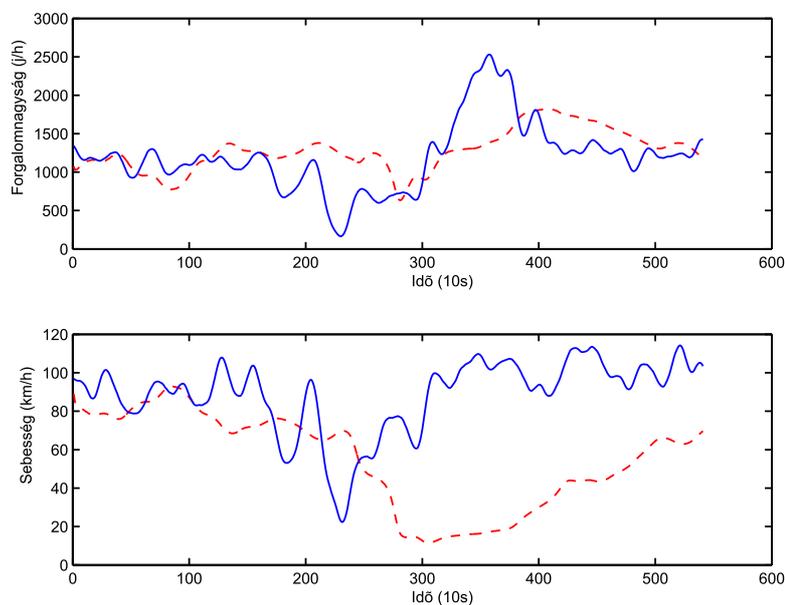
Ezek után tekintsük a rendszer választát, az állapotbecslést azonos normál forgalmi adatok mellett, mint a lineáris esetben (piros jelzi a becslült, kék pedig a valós értékeket):



4.1. ábra. Kiterjesztett Kalman-szűrő állapotbecslése normál forgalom esetén

Amint az - remélhetőleg - jól látható a nem-lineáris rendszerek esetében használható kiterjesztett Kalman-szűrő az ellenőrzőponton jól megközelíti a valós adatokat. A forgalomnagyság görbe megfelelő, míg a sebességörbe is látványosan jobb, mint lineáris esetben.

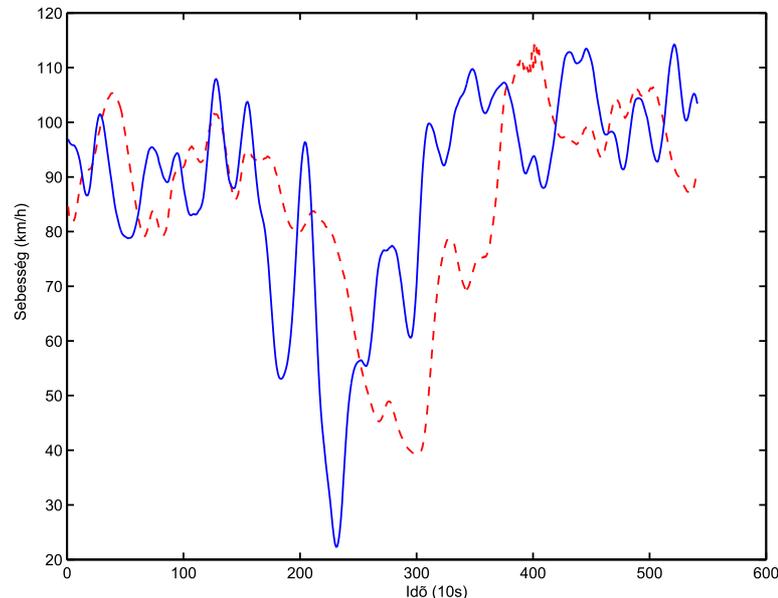
Nézzük hogyan működik az állapotbecslésünk baleset esetén, a szokásos jelölésmóddal: (piros az állapotbecslés, kék a valós adat).



4.2. ábra. Kiterjesztett Kalman-szűrés baleset esetén

Amint az jól látható a Kiterjesztett Kalman-szűrővel a forgalom lefolyás változása jobban követhető, azonban még egy fontos problémát kell megoldanunk.

Az ábrát tekintve látszik, hogy a baleseti sebességcsökkenést az állapotbecslő követi, azonban a baleset utáni sebességnövekedést nem képes megfelelően követni. Ennek oka a már említett homogén forgalom esetén érvényes 2.3 egyenletből fakad, hiszen ez inhomogén esetben érvényét veszti. Ennek a megoldására a zajok identifikációjára vonatkozó ún. adaptív szűrés jelentene kielégítő problémát, illetve az inhomogén esetben megjelenő zaj megfelelő modellezése. Az éles dinamikaváltás során a zaj értéke is megnő. Ennek a problémának közelítő megoldására a következőket hajtottam végre. Az eredetileg zajtalan forgalomsűrűség egyenleteihez egy kis zavaró hatást vettem figyelembe. Ennek a fiktív folyamatzajnak értékét úgy választottam meg, hogy a modell normál forgalomlefolyás esetén még pontos eredményt adjon, míg inhomogén forgalmi körülmények mellett jobban tudjon alkalmazkodni a fennálló viszonyoknak. Csupán a forgalomsűrűség "zaj" felvételével azonban még nem adódik közelítő megoldás, ehhez a másik két változó folyamatzajának módosítását is el kellett végezni. Így már jobb megoldást mutat a szűrő az ellenőrzőponton (pirossal a becslés, kézzel a valós adatok):



4.3. ábra. Kiterjesztett Kalman-szűrő állapotbecslése baleset esetén

Mint az látható, a sebesség már közel megfelelően képes követni a forgalom ingadozását, mind csökkenő, mind növekvő értékek esetén, igaz a követés sebessége nem megfelelő. A sebesség jobb követésének "ára" ez esetben a forgalom nagyság görbe torzulása¹. Ennek magyarázata a folyamatzajokban jelentkező változásokból fakad. A dinamika hirtelen változása esetén - esetünkben balesetkor - a folyamatzaj ugyanis hirtelen megugrik. Hangsúlyozom, hogy precíz modellezéssel (rendszeridentifikáció elvégzése), adaptív (alkalmazkodó) szűrővel megoldható a probléma, de ez már nem képezi a diplomatervem témáját. Ugyanakkor a célkitűzés szempontjából nem sérül a felállított eljárás, ugyanis a forgalomsűrűség értéke baleset

¹ Itt jegyezném meg, hogy a szakasz négy kilométeres hossza is erősen korlátozza az állapotbecslés minőségét. Az állapotbecslést 2,5 kilométeren is elvégezve ugyanis pontosabb eredményeket értem el.

esetén megnő az átlagsebesség értéke pedig lecsökken, mely változásokat az eljárással megfelelően tudunk becsülni. Így ezen változók állapotbecslésére alapozhatjuk az AID rendszer működését.

Összefoglalva elmondhatom, hogy a kiterjesztett Kalman-szűrő segítségével a forgalomlefolyás állapotbecslése megfelelő minőségű a sebesség karakterisztikát figyelembe véve. A fennálló probléma megfelelő színvonalú megoldására vonatkozó eljárásokról illetve egyéb továbbfejlesztési lehetőségekről bővebben a fejezet végén lesz szó.

4.3. Paraméterek becslése kiterjesztett Kalman-szűrővel

Az eddigi lineáris és nem-lineáris állapotbecslés esetén a vizsgált négy kilométeres szakaszon konstans modell-paramétereket feltételeztünk, azonban ez nem biztos, hogy igaz. Számos külső körülmény befolyásolhatja ezeket az értékeket, így azok az egyes szegmensekben eltérőek is lehetnek. Ebből kifolyólag az "off-line" kalibráció - amellet, hogy időigényes feladat - által megállapított értékek felhasználásával az állapotbecslés nem lehet annyira pontos, mint ha differenciálnánk szegmensekként az értékeket. Ebből kiindulva, a pontosabb eredmény érdekében a kiterjesztett Kalman-szűrő algoritmusát felhasználhatjuk a modell egyes paramétereinek becslésére is.

Ehhez a már felépített szegmensmodellt módosítottam. Az ismeretlen paraméterek között v_{free} -t és ρ_{cr} -t becsültem meg, megengedve hogy értékük térben és időben változó legyen. Ennek érdekében a paramétereket az egyes szegmensekben mint "társ-állapot" vettem figyelembe, melyeket a következő egyenletek írnak le:

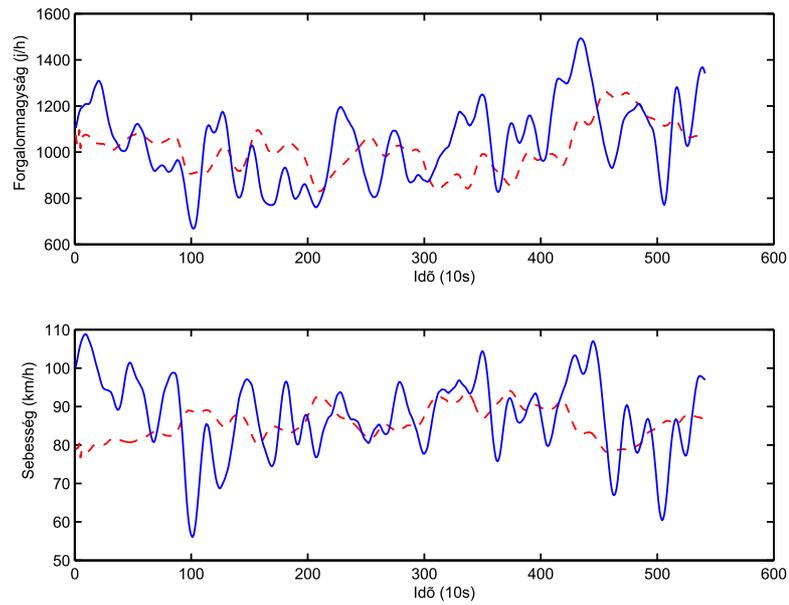
$$v_{free}(k+1) = v_{free}(k) + \xi_{v_{free}}(k), \quad (4.14)$$

$$\rho_{cr}(k+1) = \rho_{cr}(k) + \xi_{\rho_{cr}}(k), \quad (4.15)$$

ahol $\xi_{v_{free}}(k)$ és $\xi_{\rho_{cr}}(k)$ jelöli a modellparaméterek bizonytalanságát. Ezzel tehát lehetővé válik két fontos modellparaméter "on-line" kalibrációja.

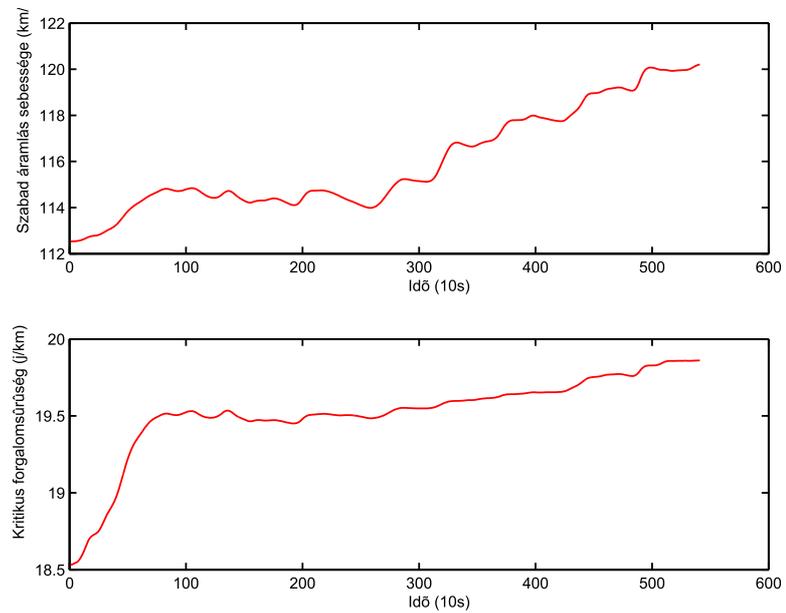
Módosítanom kellett a már felépített modelleket és programokat, hogy alkalmasak legyenek ezen paraméterek becslésére is. A becsülni kívánt modellparamétereket v_{free} -t illetve ρ_{cr} -t változónak tekintettem, melyeket mint bemenőjel kapja meg a modell. Az aktuális becsült érték körüli linearizálást kibővítettem, az aktuális munkapontban megjelenik a szegmensbeli aktuális becsült paraméterérték. Majd a paraméterek értékét mint kimenőjel csatoljuk vissza a rendszer bemenetére. Az így felépített szegmenst az B.5 SIMULINK ábra mutatja.

Mivel a paraméter becslés során két új állapotot vezettem be a modellbe, így az ezeket terhelő folyamatzajokat $\xi_{v_{free}}$ -t illetve $\xi_{\rho_{cr}}$ -t is figyelembe kell venni a W kovariancia mátrixban. Értékükre a Γ mátrix megfelelő értékeit a többi folyamatzajhoz képest kisebbnek választottam. A két változó szegmensbeli kezdeti értékeinek $v_{free} = 120 \frac{km}{h}$ -t illetve $\rho_{cr} = 20 \frac{j}{km}$ -t állítottam be. Tekintsük a rendszer válaszát az ellenőrzőponton (piros színnel az állapotbecslés, kézzel a valós adatok):



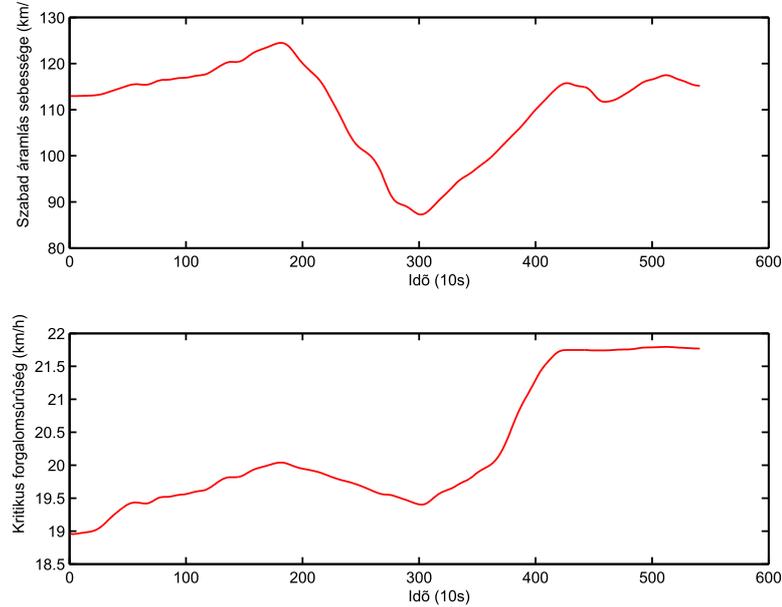
4.4. ábra. Paraméterbecsléssel bővített kiterjesztett Kalman-szűrő állapotbecslése és a valós adatok az ellenőrzőponton

Ezek mellett tekintsük az adott szegmens becsült paramétereit is:



4.5. ábra. A becsült paraméterek változása az ellenőrzőponton

Jól látható, hogy időben a szegmens szabad áramlási sebessége, illetve a kritikus forgalm nagyság változik. Ezzel tehát jobb alkalmazkodó képességet értünk el. Nézzük hogyan változik e két fontos paraméter értéke baleset esetén:



4.6. ábra. Modell paraméterek változása baleseti körülmények esetén

Az ábráról világosan leolvasható, hogy abnormális esemény esetén a paraméterek értéke hirtelen változni kezd. Arra, hogy mit jelent ez a változás a következő, az út kapacitására vonatkozó összefüggés ad választ:

$$q_{cap}(v_{free}, \rho_{cr}, a_m) = v_{free} \cdot \rho_{cr} \cdot \exp\left(-\frac{1}{a_m}\right). \quad (4.16)$$

Vagyis baleset esetén v_{free} illetve ρ_{cr} értéke csökken, ami azt jelenti, hogy az út kapacitása lecsökken, tehát az állapotbecslésünk helytálló. A baleset megszűnése utáni időszakban a paraméterek értéke emelkedik, vagyis ez jelzi, hogy az út kapacitása ismét teljes értékű, a haladás zavartalan. További észrevétel, hogy a szegmens paraméter-értékeit tekintve látható, hogy csak kis mértékben csökkentek azok, ebből arra lehet következtetni, hogy a szegmenstől "távol" történt a baleset.

Összefoglalva a paraméterbecsléssel kiegészített állapotbecsléssel lehetővé vált a szegmensek térbeli és időbeni differenciálása, ezáltal jobb állapotbecslést illetve fontos forgalmi információkat kaptunk, melyeket az Automatikus Esemény Detektálásnál felhasználhatunk.

5. AUTOMATIKUS ESEMÉNY DETEKTÁLÁS A BECSÜLT FORGALMI VÁLTOZÓK FELHASZNÁLÁSÁVAL

A fenti eredményeket összefoglalva elmondhatom, hogy a kidolgozott állapotbecslő eljárás(ok) segítségével a forgalom lefolyásáról jó becslés adható a detektor állomások között. Ezek után azt vizsgáltam, hogy milyen kritériumok alapján határozzuk meg a becsült forgalmi változók értékéből, hogy az adott szegmensben történt-e valamilyen abnormális jelenség, vagy pedig normál, zavartalan forgalom áll fenn. Másszóval elérkeztünk az Automatikus Esemény Detektáláshoz. Az első részben bemutatom a detektálás működési elvét és a felhasznált módszert. Második részben ismertetem a sebességgradiens módszert konkrét, valós baleseti adatok esetére. Végül a rendszer értékelését mutatom be és a hiányosságok megoldására szolgáló továbbfejlesztési lehetőségeket mutatom be.

5.1. *Sebességgradiens módszer*

A legegyszerűbb megközelítés szerint baleset - vagy bármilyen incidens, esemény - esetén a helyszín előtt a járművek feltorlódznak. Ez a forgalmi változók nyelvén azt jelenti, hogy az átlagsebesség hirtelen lecsökken, míg a forgalomsűrűség megnő, hiszen az út kapacitása lecsökken, a továbbhaladás nehezebbé válik. Természetesen ennek az értékváltozásnak a mértéke változó, számos tényező függvénye. Alacsony forgalomnagyság esetén egy külső sávban történő baleset nem okoz a forgalomban akkora mértékű változást mint például egy közepes- vagy nagy forgalomnagyság mellett bekövetkező akár egy, akár két sávot elzáró baleset. További problémát jelentenek a visszatérő torlódások. Ezeknél az átlagsebesség folyamatosan, simán csökken le, míg a forgalomsűrűség ezzel azonosan folyamatosan nő. Ezeket a megfontolásokat tartottam a legfontosabbaknak, amikor az Automatikus Esemény Detektálást valósítottam meg.

Statikus határértékekhez való viszonyítás esetén a rendszer képtelen az aktuális forgalomhoz alkalmazkodni és ezáltal az eredmény nagy mértékben romlana. Ezért mindenképpen dinamikus feltételekkel szükséges dolgoznunk. Ennek megvalósítására dolgoztam ki a következő eljárást, melyet sebességgradiens módszernek neveztem el.

Az alapötlet igen egyszerű, az állapotbecslés eredményeit felhasználva kiszámítjuk a függvény(ek) meredekségét és amennyiben ez a változás, illetve annak mértéke meghalad bizonyos szintet akkor jelez balesetet a rendszer. A másik fontos kiindulási pont pedig az volt, hogy nem az egyes szegmensek értékeit vizsgálom időben, hanem a szegmensek értékeit egymáshoz viszonyítva. Ezzel az eljárással - mint azt később majd látni fogjuk - kiküszöbölhető a visszatérő torlódások okozta riasztások, továbbá a szakasz forgalmi viszonyairól többet tudhatunk meg, hiszen egységként vizsgáljuk őket.

A probléma a "meredekség" meghatározásánál, hogy nem áll rendelkezésünkre konkrét függvénykapcsolat, amit differenciálva megkaphatnánk a görbe érintőjét az egyes pontokban. Viszont rendelkezésre áll a Kalman-szűrő eredményeként kapott függvény értékek. Ilyen esetben numerikus deriválással megoldható a feladat, a következő módon. Numerikus deriválás alatt olyan differenciálást értünk, amikor nem áll rendelkezésünkre $f(x)$ függvénykapcsolat, viszont az egyes $f(x_i)$ függvényértékek igen. Ilyenkor egy $p(x)$ interpolációs segédpolinommal közelíthetjük a függvényünket, mely deriválását már el tudjuk végezni. Az x_i értékeket nevezük alappontoknak, célszerű x skáláját ekvidisztánsnak választani.

Forgalmi változók, illetve adatok esetében a numerikus gradienst a következő képpen számoltam. Az adott időpontban rendelkezésre álló sebességértékeket figyeltem meg. Vagyis a vizsgált értékek száma 9, hiszen a szakasz elején és végén rendelkezésre álló valós adatokat is vizsgáltam, nem csak a Kalman-szűrő által becsült köztes állapotokat (7 darab sebességérték). Ha most kiszámolnánk az adatok numerikus gradiensét, akkor a gradiens vektorok a legnagyobb érték irányába mutatnának. Azonban mi arra vagyunk kíváncsiak, hogy a nyolc szegmens közül hol esik le a sebesség értéke. Ennek érdekében a sebesség értékek mínusz egyszerűsítését vettem, így a legnagyobbból lett a legkisebb és fordítva. Ezek után már kiszámolhatjuk a numerikus gradienst, hiszen a vektorok így a legkisebb érték felé mutatnak majd.

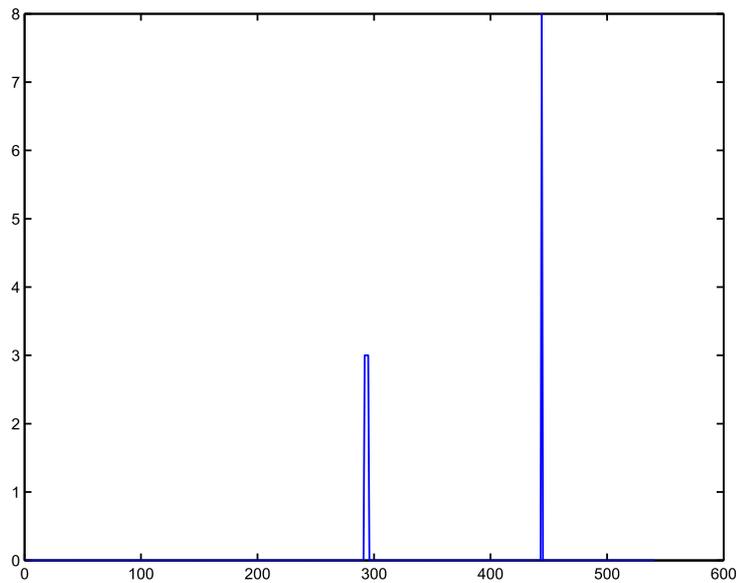
Miután így módon meghatároztam a sebesség-gradienst, azt vizsgáltam, hogy hol, melyik szegmensben található a legkisebb sebességérték. Ennek megtalálásakor azzal a feltételezéssel éltem, hogy a vizsgált szakaszon egy kitüntetett sebességminimum található baleset esetén, ott ahol a baleset bekövetkezett, hiszen először ott csökken a járművek haladási sebessége. Továbbá azt is feltételeztem, hogy a szakaszon egyidőben csak egyetlen helyen történik baleset, amennyiben egyáltalán történik. Ennek a pontnak a megtalálása igen egyszerű, hiszen amennyiben a szegmens elején illetve végén érvényes gradiens-vektorokat összeszorozzuk, úgy amennyiben ez az érték kisebb (vagy egyenlő) mint zérus, úgy a szegmensben van a "szélsőérték" a sebességértéknek. Ennek magyarázata, hogy a gradiensvektorok a kisebb érték irányába mutatnak, amennyiben a "szegmensben" előjelet vált a gradiensvektor, úgy a szegmensben kell lennie a szélsőértéknek. Ahhoz, hogy valóban a minimális értéket kapjuk - és ne a maximálisat - azt a feltételt kell előírni, hogy a szegmens elején nagyobb legyen a gradiens értéke mint a szegmens végén.

Amennyiben előjelváltást nem talál a program, úgy a baleset helyét 0-nak jelzi, vagyis nincs baleset. Viszont amennyiben talál ilyen kitüntetett pontot, úgy a szegmens számával (1-től 8-ig) jelzi, hogy ott minimum van. Elképzelhető azonban, hogy talál ilyen minimum pontot, de az nem baleset miatt áll fenn. Ennek kiszűrése érdekében azt vizsgáltam, hogy mekkora a sebességgradiens vektorok összege. Ahhoz, hogy ne kapjunk hibás adatokat további vizsgálat szükséges. Két határértéket állítottam be, annak megfelelően, hogy a szakasz közepétől jobbra vagy balra talált eseményre jellemző szélsőértéket. Ha a szakasz közepétől balra, úgy a határérték negatív, hiszen ez mutatja, hogy összességében a gradiensvektorok negatív irányban állnak, míg ha a negyedik szegmenstől jobbra talált szélsőértéket, úgy a határértéknek pozitívnak kell lennie. Amennyiben a gradiensvektorok összege egy bizonyos határértéket átlép, úgy azt baleseti lehetőségnek jegyzi fel a rendszer. Ezáltal minden időpontban meghatározható, hogy hol lehetséges baleset. Ahhoz, hogy valóban riasszon a rendszer azonban további feltételek szükségesek. Ez a már említett esemény tartósságának vizsgálata, vagyis mennyi ideig áll fenn a baleseti körülmény. További feltételként előírtam, hogy az eredő gradiensvektor abszolútértéke növekedő tendenciát írjon le. Tehát, ha a rendszer 1,5 percig ugyanazon helyen jelzi baleset megtörténtét csak akkor ad riasztást a szegmens számát kijelvezve. Az algoritmust a A.8 fájl tartalmazza.

5.2. Esettanulmány

Az általam vizsgált adatsorban található mint azt már láttuk, történt egy baleset. Nézzük, hogyan működik az sebesség-gradiens módszer valós esetben!

A már ismertetett kiterjesztett Kalman-szűrő állapotbecslése jelentette a vizsgálandó adatsort. A fent vázolt eljárás során a baleseti lehetőségeket folyamatosan visszacsatoltam. Az adott lépésben kapott értéket a vektor végére helyeztem, míg az első adatot (1,5 perccel korábban) elvettem. A 1,5 órás szimuláció során az AID rendszer eredménye a következő ábrán látható. A vízszintes tengelyen az idő, míg a függőleges tengelyen a jelzett szegmensek száma található.

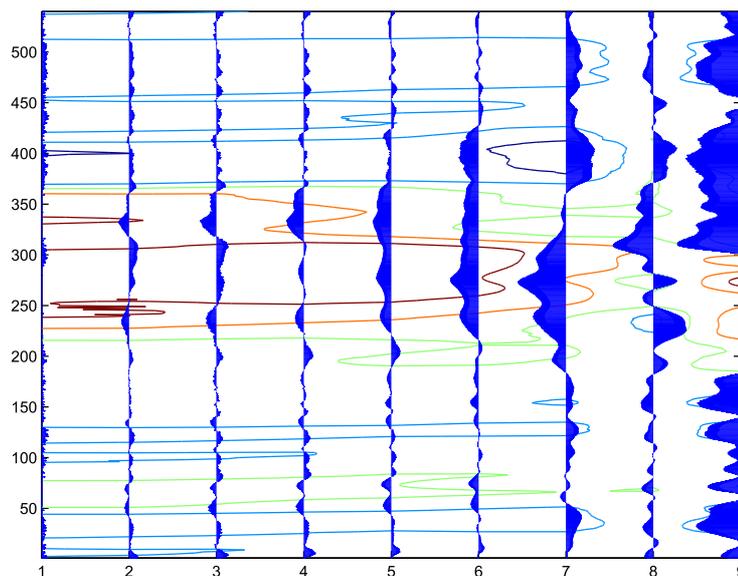


5.1. ábra. Automatikus Esemény Detektálás sebesség gradiens módszerrel

Ahogy az jól látható a rendszer két ízben jelzett balesetet, egyszer a 3-ik, másodsor a 8-ik szegmensben. A vizsgált adatsorban a 3-ik szegmensbeli baleset a valós, tehát a 8-ik szegmensben történt riasztás téves. **Azonban, ami a diplomaterv célja volt, sikeresen megtörtént, vagyis a felépített rendszer eredményesen detektálta a bekövetkezett balesetet!**

5.3. Értékelés

További fontos információkat kaphatunk, ha a sebesség-gradiens eljárást "off-line" vizsgáljuk, mintegy visszaellenőrizve a működést. A szimuláció során rögzített adatokat dolgoztam fel. Azzal, hogy rendelkezésre áll a térben és időben lezajló folyamat adatsora lehetőség nyílik a sebességgradiens módszer további vizsgálatára. Ennek során a numerikus gradiens értéket nem csak térben, hanem a szegmensek esetében időben is számítani lehet. Ezzel előáll a sebességgradiens térképe a vizsgált folyamatnak, melyet a következő ábra szemléltet:



5.2. ábra. A vizsgált folyamat sebességgradiens térképe

A vízszintes tengely a térbeli elhelyezkedésnek megfelelő az egyes számok a szegmensek kezdetét jelölik, míg a függőleges az idő tengely. A kék "hullámok" mutatják a szegmensekben a sebességgradiens értékét és irányát. A színes görbék pedig a szegmensek sebességeit tartalmazó mátrix szintvonalait szemléltetik. Ennek esetünkben különösebb jelentősége nincs. Az elsőre kissé összetett ábrát jobban megvizsgálva láthatjuk, hogy az időtartományban mikor intenzívebb a vektorok irányítottsága és mértéke. Normál áramlás esetén kiegyensúlyozottabb, nincs kitüntetett irány. Ilyen normál forgalmat feltételezhetünk a 0-200 lépésközbeli tartományon, ahol mint látható nincs kiugró intenzitású sebességgradiens. Azonban baleset esetén a vektorok intenzitása megnő (sebesség-különbség miatt) és irányítottságuk a baleset felé mutat. Az ábrán látható, hogy 230-300 lépésköz között áll fenn ilyen eset. Még jobban megvizsgálva leolvasható, hogy a gradiensvektorok intenzitása a kettes-hármas szegmensben a legkisebb ebben az időtartományban. Továbbá a 250-300 tartományban a vektorok egyértelműen a harmadik szegmens felé mutatnak, jelezve nekünk a minimumot, vagyis a baleset valóban a harmadik szegmensben történt. Az idő továbbhaladtával az vektorok intenzitása lecsökken, nincs kitüntetett irány, a forgalom ismét normál lefolyású. Az on-line Automatikus Esemény Detektálás helyességéről off-line is megbizonyosodhattunk a sebességgradiens módszer segítségével.

Az on-line és az off-line eredményeket tekintve a 8-es szegmensben történtek magyarázata. Mint látható, a 8-as szegmens végén található mérőhely értékei nagy mértékben torzítják az adatokat, ez jól látható mindkét esetben. Ennek kiküszöbölése az állapotbecslés javításával megoldható.

Összefoglalva az eredményt: a kifejlesztett eljárással - a hiányosságok mellett is - lehetőség nyílt az állapotbecslés eredményeit felhasználva azonosítanunk az autópálya-szakaszon történt balesetet. A rendszer működéséhez még számos fejlesztést szükséges elvégezni, de az alapelveket sikerült tisztázni, továbbá választ kaptunk olyan fontos kérdésekre, hogy a már meglévő rendszer hiányosságai honnan származnak illetve ezek megoldását miként lehet elvégezni.

5.4. Továbbfejlesztési lehetőségek

Rendszeridentifikáció

A lineáris Kalman-szűrés illetve az EKF esetében is javasolt egy rendszeridentifikációs eljárás során megállapítani a valódi paraméter értékeket. Ezáltal a hangolás után pontosabb állapotbecslést érhetnénk el.

Paraméterbecslés

A jobb eseményérzékelés érdekében kívánatos lehet a jövőben a modell ismeretlen paramétereinek "on-line" becslése. Ez kiterjedne az eddig nem becsült paraméterekre is, ezáltal hitelesebb modell állna rendelkezésre és így az eseményérzékelés is megfelelőbb színvonalú lenne. A rendszer alkalmazkodó képességét is javítaná az így megvalósított Kalman-szűrés.

Adaptív szűrés

A rendszer minőségének legsarkalatosabb pontja a zajok kovarianciájának meghatározása. Erre nyújtana megoldást az ún. adaptív szűrés alkalmazása, mely a kiterjesztett Kalman-szűrő algoritmusának a zaj kovarianciákra vonatkozó frissítését is tartalmazó technikája. Az általam felállított modellben a folyamatzajra illetve a szenzorzajra számos feltételezést igaznak tekintettem, melyeket a valós esetekben módosítani kellene. Az egyik feltételezés szerint a zajok kovarianciája W illetve V értékei állandóak. Ezek a dinamika változása során, illetve a környezeti viszonyok változásakor azonban változhatnak. Például a mérési zaj megnövekedhet rossz környezeti viszonyok esetében (pl. eső, köd, hó). Másik szempontból pedig a folyamatzaj is megnövekszik a hirtelen dinamika váltás során. Mint láthattuk a hagyományos technikák ezt a jelenséget csak csökkentett képességben képesek modellezni. Ilyen esetekben a Kalman-szűrő algoritmusát kibővíthetjük a zajok kovarianciájának becslésével. Ezáltal automatikusan "frissítődnek" (*Covariance Update*) ezek az értékek, így a szűrő jobban alkalmazkodik a körülményekhez, az állapotbecslés minősége javul. Ez az ún. adaptív szűrés folyamata, mely a zajok identifikációjára vonatkozik [11].

MHE - Moving Horizon Estimation

Moving Horizon Estimation, vagyis Mozgó Ablakos Becslés. Ez az eljárás az állapotbecslés során nem csak az egy lépéssel előző időpontbeli értékeket használja fel az állapotbecsléshez, hanem egy időablakot képezve több lépés eredményeit. Továbbá fontos előnye, hogy megadhatók korlátozó feltételek is (constrain MHE), melyeket felhasználva az állapotbecslés

"nem-megengedett" értékei kiküszöbölhetőek. Nem-lineáris esetben is alkalmazható, megbízható eljárás, mely alkalmas lehet a jövőben az autópálya forgalmi változóinak becslésére és így az AID rendszer alapjául szolgálhat [14], [15].

LPV - Linear Parameter Varying

Az LPV technika speciális modell-egyenletek esetén használatos, ahol az egyenletekben szereplő paraméterektől függően a rendszer nem-lineáris, illetve a vizsgált paraméterek fontos szerepet játszanak a rendszer dinamikájában. Az ilyen rendszerek analízisére és szintézisére kidolgozott technika az elért eredmények tükrében alkalmazható lenne az AID (illetve állapotbecslő) rendszer esetén.

AID rendszer fejlesztése

A sebességgradiens módszer alkalmasnak bizonyult a balesetek detektálására. Azonban a rendszer megfelelő működéséhez még - a modell és az állapotbecslés javításán túl - módosításokra van szükség. Ennek során felhasználhatóak lehetnének a rendelkezésre álló további forgalmi változók, illetve az on-line kalibráció eredményei. A rendszer "érzékenységének" javítása érdekében a dinamikai feltételek jobb kidolgozása, illetve a rendszer megbízhatóságának érdekében a feltételek felülvizsgálata szükséges. Külön figyelmet kell fordítani a későbbiekben arra, hogy az eljárás minden forgalomnagyság és lehetőleg minden baleset érzékelésére alkalmas legyen.

ÖSSZEFOGLALÁS

Diplomatervemben az autópálya forgalom dinamikus modelljén alapuló állapotbecsléssel megvalósított Automatikus Esemény Detektáló rendszert dolgoztam ki.

Ismertettem az autópálya forgalom modellezésének szempontjait, a modell paramétereinek jelentését és ezen paraméterek kalibrációjának menetét. A modell egyenletek alapján, a rendelkezésemre álló valós helyszínt modelleztem le, melyet a szakasz detektorállomásainak adatai alapján kalibráltam.

Az így kapott modell esetén először lineáris Kalman-szűrőt terveztem, mely a forgalom "normál" lefolyása esetén megfelelően becsülte a forgalmi változókat. Bemutattam a lineáris Kalman-szűrő tervezésekor fontos frekvencia-tartománybeli analízis lényeges szempontjait és lépéseit, azok alkalmazását a konkrét autópálya-szakasz esetében. Bemutattam a lineáris Kalman-szűrő hiányosságait és ezen hiányosságok megoldására javaslatot tettem.

A lineáris Kalman-szűrő tervezését követően nem-lineáris, kiterjesztett Kalman-szűrővel is megoldottam a forgalmi paraméterek becslését. Majd a modell ismeretlen paramétereinek közül két - fontos - paraméter "on-line" becslését is elvégeztem. Ezáltal a forgalom lefolyásról megfelelő képet alkothatunk az autópályán elhelyezett detektorok adatait felhasználva.

A vizsgált rendszer rávilágított az eljárás hiányosságaira, melyek kiküszöbölésére léteznek megfelelő eljárások. Eredményül elmondhatom, hogy a jövőbeli alkalmazásoknak mindenképpen nem-lineáris állapotbecsléssel szükséges dolgozniuk, valamint a folyamatot illetve mérést terhelő zajokban bekövetkező változások kezelésére is alkalmasnak kell lenniük.

Az így kapott állapotbecslő eljárás adatait használtam fel a bekövetkező események automatikus érzékelésére. Valós adatok felhasználásával mutattam meg, miként történik balesetek azonosítása. Az adatok feldolgozására egy új - dinamikus - algoritmust írtam, mely a becsült állapotváltozóknál bekövetkező hirtelen változások esetén jelez baleseti állapotot.

Megjegyezném, hogy bár munkám során számos feltételezéssel és egyszerűsítéssel éltem, azonban remélhetőleg így is sikerült bemutatnom, miként működnek az AID rendszerek, azon belül pedig az állapotbecslést felhasználó eljárások. Ahhoz, hogy a rendszer valóban alkalmas legyen balesetek azonosítására a mindennapi használatban, még számos feladatot szükséges megoldani, ám ebből a szempontból számos új információt szereztünk.

Összefoglalva az elért eredményeket: a modern autópálya irányítás és felügyelet fontos részét képezné egy ilyen rendszer. Az általam felépített és bemutatott rendszer a későbbiekben

alapjául szolgálhat egy valós AID rendszer bevezetésének. Amint azt jól láthattuk a modell képes az autópályán bekövetkezett események érzékelésére, amellyel így csökkenthetőek az anyagi, személyi és környezeti károk. Továbbá a forgalomfigyelő rendszer adatainak felhasználásával rugalmas és megbízható autópályairányítás valósítható meg a jövőben elterjedő változtatható jelzésekű táblák segítségével.

IRODALOMJEGYZÉK

- [1] Dr. Emily Parkany, PI Chie Xie: *A Complete Review of Incident Detection Algorithms & Their Deployment: What Works and What Doesn't* The New England Transportation Consortium, Februar 7, 2005.
- [2] Dr. Peter T. Martin, Associate Professor, Joseph Perrin, Ph.D., PE, PTOE, Blake Hansen, M.S.: *INCIDENT DETECTION ALGORITHM EVALUATION* Utah, march 2001. ([http : //www.ndsu.nodak.edu/ndsu/ugpti/MPC_Pubs/html/MPC01 – 122/](http://www.ndsu.nodak.edu/ndsu/ugpti/MPC_Pubs/html/MPC01-122/))
- [3] California Center for Innovative Transportation honlap ([http : //www.calccit.org/](http://www.calccit.org/))
- [4] Michalopoulos, P.G.: *Automatic Incident Detection Through Video Image Processing* (www.autoscope.com)
- [5] Lantos Béla: *Irányítási rendszerek elmélete és tervezése I. - Egyváltozós szabályozások*, Akadémia Kiadó, Budapest, 2001.
- [6] Lantos Béla: *Irányítási rendszerek elmélete és tervezése II. - Korszerű szabályozási rendszerek*, Akadémia Kiadó, Budapest, 2003.
- [7] J.M.Maciejowski: *Multivariable feedback Design*, Addison-Wesley, 1993.
- [8] M.Papageorgiu: *Lecture Notes in Control and Information Sciences - Applications of Automatic Control Concepts to Traffic Flow Modeling and Control*, Springer-Verlag, 1983.
- [9] Greg Welch and Gary Bishop: *An Introduction to the Kalman Filter* TR 95-041 Department of Computer Science, University of North Carolina at Chapel Hill, April 5, 2004.
- [10] Dr. Bokor József: *Lecture Notes in Modern Control Theory II. PhD. course* 2006.
- [11] M.Oussalah, J. De Schutter: *Adaptive Kalman Filter for Noise Identification*
- [12] A.V.Balakrishnan: *Kalman Filtering Theory* Optimization Software, INC. - Publications Division, New York, 1984.
- [13] Kulcsár Balázs, Varga István, Bokor József: *Modern közúti forgalomirányítás I. A forgalmi paraméterek becslése* Városi Közlekedés 2005/1
- [14] Kulcsár Balázs, Varga István, Bokor József: *Constrained split rate estimation by moving horizon* 16th IFAC World Congress, July 4-8. 2005, Praha, Czech Republic, p.3276.

- [15] Varga István, Kulcsár Balázs, Bokor József: *General Moving Horizon Estimation of Traffic Systems* 12th Mediterranean Conference on Control and Automation, Kusadasi, Turkey, June 6-9. 2004, p. 1125.
- [16] dr. Tóth János, Varga István: *A korszerű közlekedési forgalomirányító rendszerek*

FÜGGELÉK

A. MATLAB SCRIPTEK

A.1. *megnyitas.m*

```
function [A,B]=megnyitas('hely')
```

A program bemenő paramétere a 'hely' string, mely a feldolgozni kívánt forgalmi adatsor helyét adja meg. A program lefutásakor a fájlból a megfelelő adatokat választja ki, továbbá időben folytonos értékek szerint rendezi az adatsort. Visszatértési értéke az A illetve B mátrix, melyek az egyes- illetve kettes sáv adatait tartalmazzák.

A.2. *kritikus.m*

```
function kritikus('hely')
```

A program bemenő paramétere a 'hely' string, mely a feldolgozni kívánt forgalmi adatsor helyét adja meg. A program az ismertett eljárás során kiszámolja az adott sáv kritikus forgalomsűrűség, szabad áramlási sebesség illetve modellparaméter értékét. A lefutás eredményeként az adatsor alapján kirajzolja a forgalomsűrűség-forgalomnagyság illetve a forgalomsűrűség-sebesség diagramokat és az értékekre illesztett görbe alapján határozható meg a kívánt paraméterek értéke.

A.3. *tsim.m*

```
function M=tsim(A,B)
```

A program bemenő paramétere a *megnyitas.m* program eredményeként kapott sávok adatait tartalmazó A illetve B mátrixok. A sávok adatait a megadott $T_s = 10s$, szimulációs lépésköz szerint bontja fel. Az adott intervallumban áthaladt járművek számát és ezeknek a járműveknek harmonikus sebességértékét számolja. Az így kapott adatokat kilencedrendű mozgóátlagos szűréssel simítja. A két forgalmi sáv értékeit összevonva kapjuk a mérőhely adatait. Az adatsort ismét simítja, az így kapott M - mérőhelyi adatokat tartalmazó - mátrix a program lefutásának eredménye.

A.4. *metanetmodell.m*

`function y=metanetmodell(u)`

A program tartalmazza a dinamikus forgalmi modell egyenleteit. A bemenő jel szintaktikája a következő:

- $u(1) = \rho_i(k)$ a szegmens forgalomsűrűsége
- $u(2) = q_{i-1}(k)$ előző szegmens kilépő forgalma
- $u(3) = q_i(k)$ szegmens kilépő forgalma
- $u(4) = v_i(k)$ szegmens átlagsebessége
- $u(5) = v_{i-1}(k)$ előző szegmens átlagsebessége
- $u(6) = \rho_{i+1}(k)$ következő szegmens forgalomsűrűsége
- $u(7) = r_i(k)$ felhajtó forgalom nagysága
- $u(8) = l_i$ szegmenshossz

A kimenőjel pedig:

- $y(1) = \rho_i(k + 1)$ a szegmens forgalomsűrűsége
- $y(2) = v_i(k + 1)$ a szegmens átlagsebessége
- $y(3) = q_i(k + 1)$ a szegmens kilépő forgalom nagysága

Az be- és kimenetek között a METANET modell teremt kapcsolatot.

A.5. *Kalman.m*

`function y=Kalman(u)`

A lineáris Kalman-szűrést megvalósító program. Bemenetei:

- $x_{po}(k - 1)$ az állapotbecslés értékei az előző időpillanatban
- $P_{po}(k - 1)$ a becslési hiba kovarianciája az előző időpillanatban
- $y(k)$ a mérési eredmények az adott időpontban

A lineáris Kalman-szűrő algoritmus után a függvény kimenete az adott lépésben becsült állapot és a hozzá tartozó becslési hiba kovarianciája.

A.6. *ExKalman.m*

```
function y=ExKalman(u)
```

A kiterjesztett Kalman-szűrést megvalósító program. Be- és kimenete megegyezik a lineáris Kalman-szűrést megvalósító *Kalman.m* függvényével. A program munkapont körüli linearizálás során előállítja a szegmensek adott munkapontra jellemző állapotter reprezentációját, majd ezekből az autópálya-szakasz állapotter modellét. A kiterjesztett Kalman-szűrő algoritmus után kimeneteként az állapotbecslés és a becslési hiba kovarianciájának értéke adódik.

A.7. *EstimPara.m*

```
function y=EstimPara(u)
```

A kiterjesztett Kalman-szűrő paraméterbecslésre módosított változata.

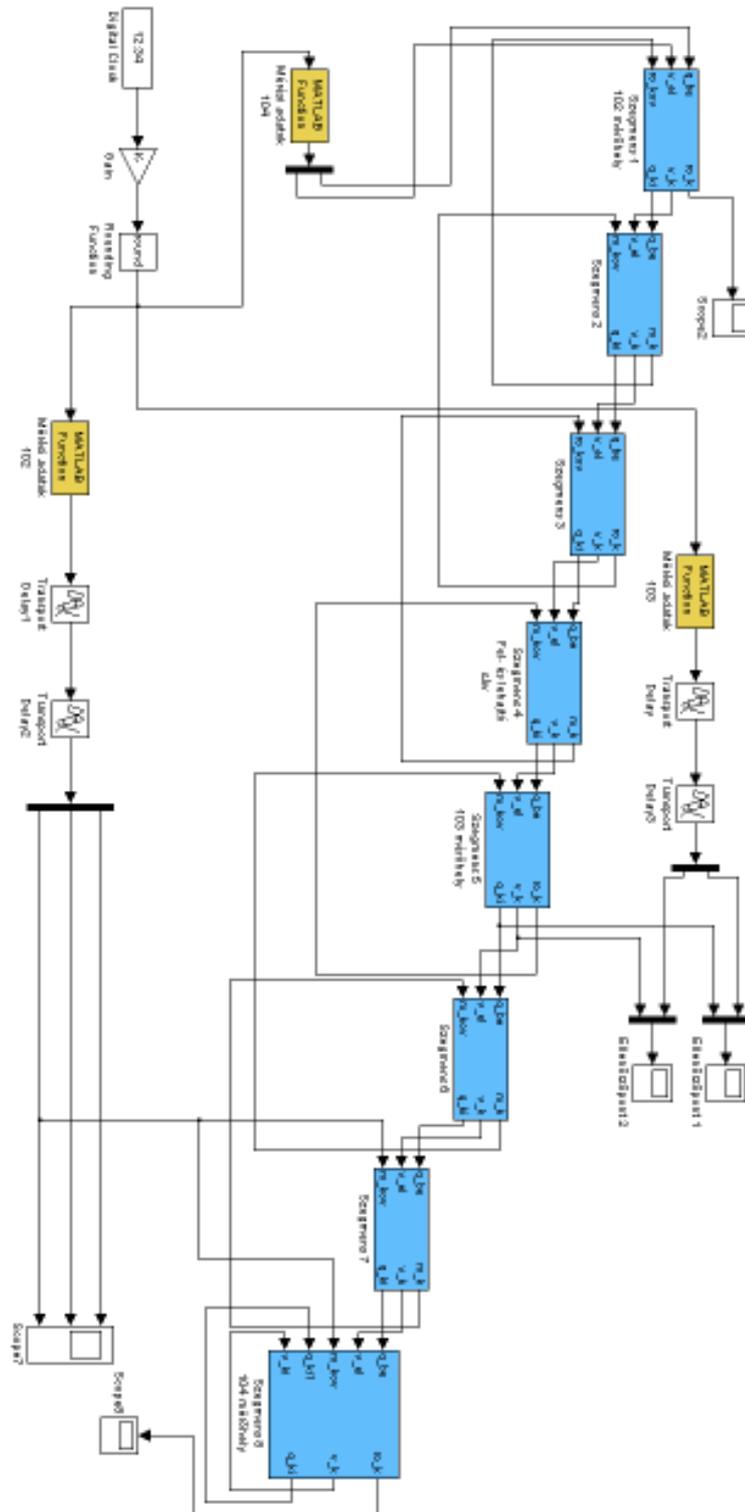
A.8. *aid.m*

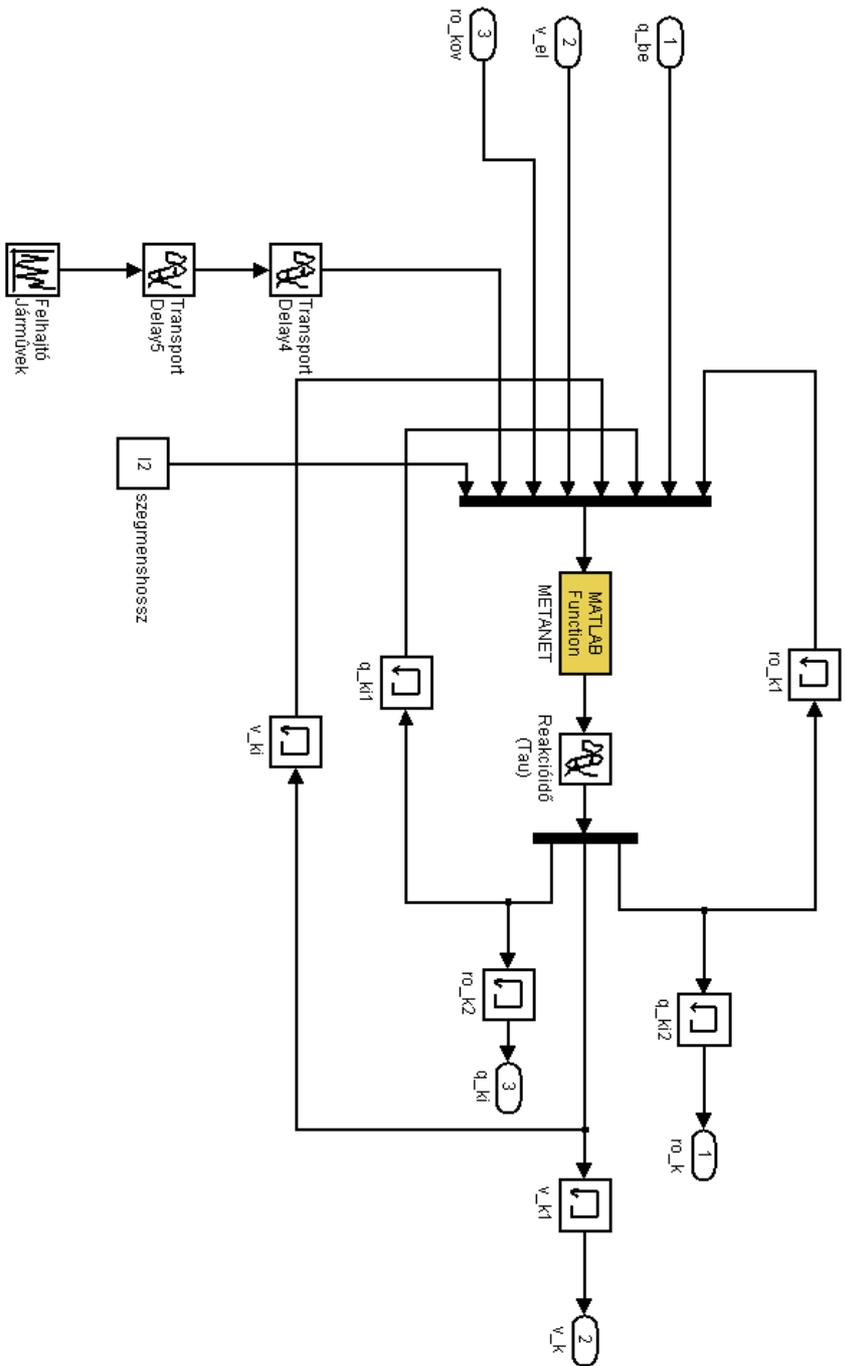
```
function y=aid(u)
```

Az automatikus esemény detektáló algoritmus, sebességgradiens módszer. Bemenőjeleként a szegmensek becsült sebességértékeit kapja meg, illetve az elmúlt 1,5 perc baleseti jelzéseit. A sebességértékek gradiensét számolja és a dinamikus feltételeknek megfelelően detektál. Kimenete az aktuális baleseti jelzés és a visszacsatoláshoz szükséges baleseti jelzéseket tartalmazó vektor.

B. MATLAB SIMULINK ÁBRÁK

B.1. ábra. Autópályaszakasz modellje SIMULINK környezetben - Metanet





B.5. ábra. Szegmens modellje paraméterek becsléséhez SIMULINK környezetben - PSzegmens

