



Közúti forgalomirányító rendszerek fejlesztése és tesztelése VISSIM-MATLAB környezetben

Tettamanti T., Varga I.

Budapesti Műszaki és Gazdaságtudományi Egyetem, Közlekedésmérnöki és Járműmérnöki Kar, Közlekedésautomatikai
Tanszék, 1111 Budapest, Stoczek utca 2.

ÖSSZEFOGLALÁS

Cikkünkben az alkalmazott informatika egy speciális ágát kiemelve mutatjuk be a VISSIM-MATLAB környezetre épülő, zárt hurkú szimulációs keretrendszer. A szimulációs környezet alkalmazása hasznos eszközként szolgálhatja közúti forgalomirányító rendszerek fejlesztését és tesztelését. A két szoftver COM és API felületen keresztüli programozhatósága nagyban megnöveli a szimulációs lehetőségeket. Ezáltal olyan kiegészítő funkciókat lehet létrehozni, amelyekkel gyakorlatilag bármilyen forgalomirányító logika megvalósítható, a bonyolultabb számítások pedig közvetlenül a MATLAB-bal elvégezethetők valós időben a forgalom-szimuláció futása közben.

(Kulcsszavak: forgalomszimuláció, forgalomirányítás, VISSIM, MATLAB, COM, API)

ABSTRACT

Development and test of road traffic control systems by using VISSIM-MATLAB environment

T. Tettamanti, I. Varga

Budapest University of Technology and Economics, Faculty of Transportation Engineering, and Vehicle Engineering
Department of Control and Transport Automation, H-1111 Budapest, Stoczek utca 2.

The paper investigates a special issue of the applied informatics. A VISSIM-MATLAB based closed loop simulation framework is presented. The application of the simulation environment may serve as a very efficient tool for development and test of road traffic control systems. The availability of these software, through COM and API programming, increase the facilities of the simulations. Therefore, it is possible to create such additional features which allow to design special traffic control logic. Furthermore, complex calculations may be done by MATLAB in real time during the run of the simulation.

(Keywords: traffic simulation, traffic control, VISSIM, MATLAB, COM, API)

BEVEZETÉS

A közúti motorizáció növekedése világtendenciát mutató jelenség. A folyamatosan növekvő járműforgalom számtalan közvetlen és externális hatást generál, amelyek áttételesen komoly társadalmi költségként jelennek meg. A probléma részleges kezelésére alkalmasak lehetnek olyan intelligens forgalomirányítási rendszerek, amelyek megfelelő adaptív stratégiával, valós időben képesek a forgalom aktuális állapotaira reagálni. Ilyen rendszerek fejlesztése és kiépítése igen komplex feladat. Ugyanakkor korunk technológiai szintjének megfelelően elvárható a teljes tervezési feladat

szoftverrel segített kivitelezése, amely során a tervezett rendszer nagy része jól tesztelhető. A tesztek segítségével hibák és számtalan előre nem látható akadályok szűrhetők ki. Ennek megfelelően ma már egyre inkább bevett gyakorlat a forgalomszimulációs szoftverrel való tervezés, tesztelés, és validálás.

Cikkünkben egy olyan szimulációs keretrendszert mutatunk be, amely hatékonyan alkalmazható a közúti forgalomirányítási feladatok, stratégiák, rendszerek fejlesztéséhez, vizsgálatához. A keretrendszer két építőeleme a VISSIM (VISSIM, 2011) mikroszkopikus szimulátor és a MATLAB (MATLAB, 2011) tudományos, matematikai szoftver, amelyeket COM (Component Object Model) interfész segítségével kapcsolunk össze.

A cikk első felében a VISSIM alapú forgalommodellezést és szimulációt mutatjuk be a forgalomirányítási logika megvalósíthatóságának szempontjából. Bemutatjuk a VISSIM COM és API (Application Package Interface) programozhatóságát, mint lehetséges eszközöket a forgalomszimuláció tulajdonságainak kiterjesztésére. A forgalommodellezés mellett kitérünk a MATLAB alkalmazásának lehetőségére, amely megfelelő API programozással szintén elérhető és felhasználható a VISSIM szimuláció futása közben.

A cikk második felében egy alkalmazási példán keresztül mutatjuk meg a bemutatott keretrendszer alkalmazhatóságát. Olyan robusztus tulajdonságú irányítási stratégiát teszteltünk VISSIM-MATLAB szimulációs környezetben, amely képes a bizonytalanságok figyelembe vételére azok előre megbecsült maximális mértéke alapján. A szimulációk hatékony eszközként szolgáltak a fejlesztett rendszer vizsgálatához.

SZIMULÁCIÓS KERETRENDSZER MODERN KÖZÚTI FORGALOMIRÁNYÍTÁS TERVEZÉSÉHEZ

A közúti forgalomirányítás folyamatos fejlesztése elkerülhetetlen a növekvő igények tükrében. Ezen belül megkülönböztetünk városi szintű, gyorsforgalmi utakra vonatkozó, és a kettő találkozási pontjain értelmezett, ún. integrált forgalomszabályozást. Az irányítási stratégiák a legegyszerűbb figyelmeztető jelzésektől (pl. balesetveszélyre való figyelmeztetés) a komplex logikát megvalósító, adaptív rendszerekig nagyon széles skálán mozoghatnak. A felhasználható szabályozó eszközök szintén sokszínű alkalmazást tesznek lehetővé: változtatható jelzésképű táblák, dinamikus sebességkorlátozás, autópálya felhajtás-korlátozás, hagyományos háromfogalmú jelzőlámpák, intelligens fedélzeti rendszerek, útvonalajánlás.

Egyszerűbb irányítások esetén előzetes mérési adatok alapján könnyen kialakítható a szükséges szabályozási stratégia, pl. rögzített idejű jelzőlámpás vezérlés. Ugyanakkor modern, irányító rendszerek esetén elkerülhetetlen az informatikai eszközök komolyabb igénybevétele.

Forgalommodellezés és forgalomirányítás VISSIM szimulátorral

A forgalommodellezés kiemelkedő fontosságú új stratégiák tervezésekor. A felhasználók ma már számtalan közúti forgalomszimulációs szoftver közül válogathatnak. Ezek legnagyobb része kereskedelmi szoftver (teljesség igénye nélkül néhány: AIMSUN, PARAMICS, TRANSMODELER, VISSIM), illetve elérhetők nyílt forráskódú, ingyenesen használható szimulátorok is, amelyeket egyetemek vagy kutató intézetek fejlesztenek (pl. MITSIM, SUMO). A cikkünkben bemutatandó keretrendszerben a forgalomszimulációt a VISSIM szoftver végzi.

A VISSIM mikroszkopikus forgalommodellezést valósít meg, ami azt jelenti, hogy a járműviselkedés individuális megközelítésén alapul. A mikroszkopikus szintű

modellezés célja a közlekedés dinamikájának széleskörű és pontos leírása. A szimulált közlekedési hálózat működése így részletesebben vizsgálható, és a képi megjelenítése is valósághűen megvalósítható. A VISSIM az ún. pszicho-fizikai járművezető-viselkedési modellre épül, amelynek elméleti háttere a Wiedemann járműkövetési modell (Wiedemann, 1974). A modell alapvető koncepciója az, hogy a gyorsabban haladó jármű vezetője akkor kezd lassítani, amikor az egyéni érzékelési küszöböt eléri az előtte haladó, lassabb jármű percepciója. Mivel nem tudja pontosan meghatározni az elől haladó gépkocsi sebességét, így addig lassít, amíg egy másik érzékelési küszöböt el nem ér, és újra gyorsítani nem kezd. A szimuláció tehát a lassulás-gyorsulás kapcsolatán alapuló, folyamatos iterációs eljárás, ahol minden jármű és járművezető egyéni tulajdonságokkal rendelkező objektum. A VISSIM ma már széles körben elterjedt szoftver, amely hiteles referenciát jelent bármilyen közúti irányítási feladat megvalósítása során (Takács, 2004).

A VISSIM sokcélú felhasználást tesz lehetővé a forgalommodellezés területén. Ugyanakkor cikkünkben kifejezetten a forgalomirányítási feladatok szimulációjára fókuszálunk. A szoftver több lehetőséget is nyújt a szabályozó logika, azaz a forgalomirányító berendezések szimulációjára. Rögzített idejű vezérlés esetén az ún. *Fixed time* jelzésvezérléssel egyszerűen definiálható a forgalomtechnika. Saját, forgalomfüggő logika megvalósításhoz a VAP modult (Vehicle Actuated Programming) lehet használni. A VISVAP tervező szoftverrel folyamatábraként valósítható meg bármilyen szabályozás, amihez természetesen a hálózatban tetszőlegesen felvett hurokdetektorokat is fel lehet használni. Az így elkészített logikát a VISVAP egy vap kiterjesztésű fájlba exportálja, ami pedig közvetlenül meghívható a VISSIM-ben. További lehetőségként választható az ún. *External* jelzésvezérlés, amely az előzőhöz hasonlóan tetszőleges, saját logika megalkotására ad lehetőséget. Ugyanakkor mindezt már C++ programnyelven kell elkészíteni a VISSIM által nyújtott Signal Controller API segítségével, ami gyakorlatilag a szabályozó forráskód-szintű programozását jelenti. A C++, mint objektumorientált nyelv magas szintű programozási lehetőséget nyújt, bőven túllépve a grafikus VISVAP alapú megoldás korlátain. Az így elkészített jelzési programot dll kiterjesztésű fájlba kell lefordítani, aminek elérési útját a vap fájlhoz hasonlóan a felhasználó adja meg a VISSIM grafikus felületén keresztül. Ezekben a megoldásokon túl a VISSIM - alkalmazkodva a piaci igényekhez - további, kereskedelmi célú szabályozási modulok alkalmazását is teszi lehetővé, pl. SCATS, SCOOT, SIEMENS VA, LISA+OMTC.

VISSIM forgalomszimuláció lehetőségeinek kiterjesztése COM felület segítségével

A forgalomszimuláció magasabb szintű irányításához a szoftvergyártó a COM felület programozását ajánlja, amihez részletes dokumentációval is segíti a felhasználót. A COM a Microsoft által kifejlesztett technológia a komponens alapú fejlesztés támogatására, mely a szoftverek közti kommunikációt teszi lehetővé. Bár több platformon is megvalósították, elsősorban a Microsoft Windows operációs rendszerében használják.

A VISSIM COM hierarchikus felépítésű, objektumorientált modellt valósít meg, amiben kódszinten válnak elérhetővé és programozhatóvá a grafikus felületen nyújtott funkciók és paraméterek. A COM felületet lehet script programnyelveken (pl. Visual Basic Script, Java Script), illetve objektumorientált nyelveken is programozni (pl. Microsoft Visual Studio programcsalád nyelvei, Java, Delphi, C++). A programozás első lépéseként egy COM klienst kell létrehozni a választott nyelven. Majd céltól függően a forgalomszimulációs folyamat egyes részeit vagy akár egészét el lehet készíteni és

automatizálni a COM-on keresztül. Amennyiben például egy saját készítésű, forgalomfüggő logikát szeretnénk kivitelezni COM-mal, a VISSIM grafikus felületén a *Fixed Time* stratégiát érdemes kiválasztani. Ekkor az irányítási logikának megfelelően - amit szintén a COM felületen készíthetünk el - a szabályozási ciklusok végén a fix idejű jelzéstervet felülírhatjuk, azaz a rögzített idejű vezérlési tervet gyakorlatilag folyamatosan frissítjük.

A gyakorlatban egy COM felülettel fejlesztett, saját logikát használó szimulációt a következőképpen érdemes felépíteni:

1. Az adott közlekedési hálózat felépítése a VISSIM grafikus felületén keresztül minden részletével együtt (útszakaszok, jelzőlámpák, detektorok, járművek, stb.);
2. COM kliens létrehozása a választott programnyelven;
3. Saját irányítási stratégia leprogramozása;
4. Speciális szimulációs beállítások leprogramozása, pl. többszöri futtatás egymásután más paraméterekkel, a járművek valamilyen feltételhez kötött viselkedése;
5. A megírt program lefordítása exe kiterjesztésű futtatható állományba;
6. A szimuláció elindítása parancssorból az elkészített exe fájl, a VISSIM projektfájl (inp) és a hozzá tartozó inicializáló fájl megadásával.

MATLAB alkalmazása VISSIM szimulációhoz

Nyilvánvaló, hogy összetettebb forgalomtechnikai stratégiát megvalósító szimulációs feladatok esetén az előző fejezetben bemutatott COM felület és az ahhoz alkalmazott magas szintű programozási nyelv nagy segítségére lehet a felhasználóknak. Ugyanakkor egyes feladatokat érdemes lehet speciális matematikai szoftverrel elvégeztetni. Egy bonyolultabb optimalizálási feladat megoldása esetén például, a felhasználónak nem feltétlen van szüksége a számítási algoritmus részletes ismeretére. Csak a módszer helyessége, és a végeredmény számít. Így egy külső szoftver által nyújtott függvény alkalmazásával, rengeteg idő és energia megtakarítható összehasonlítva a programnyelven történő megvalósítással. Mindehhez azonban természetesen alapfeltétel a matematikai szoftver elérhetősége a forgalomszimuláció futtatása alatt.

A forgalomszimulációs feladatokhoz segítséget nagy nyújthat a MATLAB program, amelyben beépített függvények százait lehet felhasználni. Robusztus mátrix műveleti csomaggal és kiváló diagramrajzoló függvényekkel rendelkezik, amelyek igen hasznosak lehetnek valós idejű alkalmazások fejlesztéséhez, ellenőrzéséhez. A MATLAB-ot a VISSIM-hez hasonlóan el lehet érni COM felületen keresztül. Ehhez a MATLAB szoftvergyártó egy MATLAB Engine nevű API-t (*MATLAB 7 External Interfaces*, 2010). bocsát a felhasználók rendelkezésére, amellyel az alábbi funkciók valósíthatók meg:

- Tetszőleges MATLAB függvény, vagy előre megírt MATLAB programkód meghívása saját programból;
- Adatok küldése/fogadása a MATLAB és saját programunk között.

Amennyiben a forgalomszimulációban ki akarjuk használni a MATLAB által nyújtott lehetőségeket az előzőekben bemutatott VISSIM COM felület programozását kibővíthetjük a MATLAB Engine API alkalmazásával. Így például dinamikus jelzéstervezéshez, diagramok rajzolásához a MATLAB alapú algoritmusok közvetlenül elérhetővé válnak.

Az VISSIM-MATLAB által megvalósított szimulációs keretrendszert az *1. ábra* reprezentálja.

1. ábra

VISSIM-MATLAB szimulációs keretrendszer

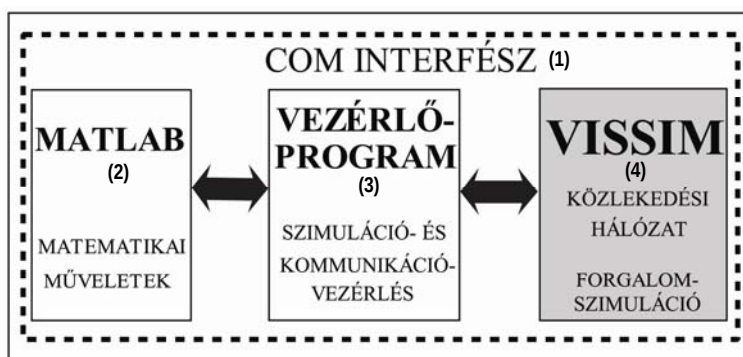


Figure 1: VISSIM-MATLAB simulation framework

COM interface(1), MATLAB: mathematical operations(2), Controller Program: Simulation and communication control(3), VISSIM: traffic network, traffic simulation(4)

A keretrendszer vezérlőprogramja természetesen bármilyen programnyelvben készülhet, ami képes COM interfész kezelésére.

VISSIM Signal Controller API programozása

Az eddig bemutatott módszerek segítségével egy nagyon hatékony tesztelési keretrendszer építhető fel. A VISSIM azonban egy további lehetőségként kínálja a Signal Controller API programozhatóságát az *External* típusú jelzésvezérlés megvalósításához, amelyet C++ programnyelven lehet elkészíteni. Ezzel a megoldással közvetlenül hozzáférhetünk a szimulációban definiált forgalomirányító berendezésekhez (*Signal Controller*) és detektorokhoz, ill. gyakorlatilag bármilyen időzítés szerint alakíthatjuk a hozzájuk kapcsolódó jelzési stratégiát. Az *Signal Controller* API segítségével képzett jelzési logikát dll fájlként kell lefordítani (külön-külön mindegyik *Signal Controller*-hez), amiket aztán a VISSIM a szimuláció során minden egyes szabályozási időtartamban (szabadon definiálható, de általában 1 másodperc) lefuttat. Először az aktuális jelzésképek és detektoradatok (minden detektor hozzá kell, hogy legyen rendelve előzőleg valamelyik *Signal Controller*-hez) érkeznek meg a vezérlő dll fájlok számára. Ezután a dll programok a beérkező információk függvényében és a felhasználó által kitalált logika alapján döntenek az új jelzésképekről, amelyek azonnal visszakérülnek az éppen futó VISSIM szimulációba. Az API segítségével az átmeneti jelzésképek is előre beállíthatók, így a frissített jelzések ennek függvényében „kerülnek ki” a jelzőfejekre. Ez a funkció igen nagy előnyt jelenthet városi forgalomirányítás szimulációja esetén, ahol így például elég a kiszámított zöldidő és pirosidő kezdeti időpillanatait megadni jelzőcsoportonként. Ugyanakkor amennyiben ugyanezt COM felületen keresztül szeretnénk megvalósítani, akkor a jelzőcsoportoknak minden egyes jelzési állapot változását meg kell adni – beleértve természetesen az átmeneti jelzésképeket is. Ez pedig nagy mértékben megbonyolítja a programozási feladatot. További előnye még az API alkalmazásának, hogy az így létrehozott jelzésvezérlés jóval gyorsabb szimulációt tesz lehetővé a COM-os megvalósításhoz képest, és az időzítése is könnyen programozható.

Ez a megoldás alkalmazható önmagában, kizárólag az *External* típusú forgalomirányító berendezések elkészítéséhez, amennyiben a felhasználó nem akar olyan speciális szimuláció vezérlést kialakítani, amit ne tudna a VISSIM grafikus felületén is beállítani. Ugyanakkor természetesen mindez kombinálható a VISSIM COM programozással. Sőt a Signal Controller API a MATLAB Engine API-val együtt is programozható a MATLAB szoftver szolgáltatásainak elérésére céljából. Az így kialakítható teljes szimulációs keretrendszert a 2. ábra mutatja.

2. ábra

A teljes szimulációs keretrendszer

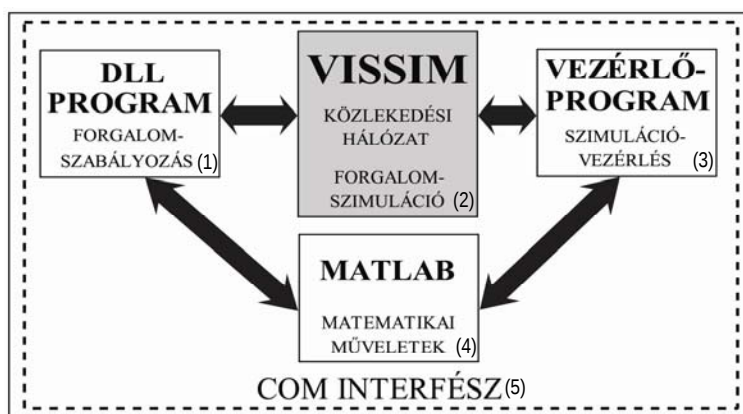


Figure 2: The whole simulation framework

DLL Program: traffic control(1), VISSIM: traffic network, traffic simulation(2), Controller Program: simulation control(3), MATLAB: mathematical operations(4), COM Interface(5)

SZIMULÁCIÓS PÉLDA

A következőkben egy alkalmazási példát mutatunk be. Az eddig leírt keretrendszer segítségével fejlesztettünk, szimuláltunk és teszteltünk egy olyan forgalomirányítási stratégiát, amely alkalmas városi úthálózatok intelligens, valós idejű irányítására.

A tesztelt robusztus városi forgalomirányítás

Az általunk tervezett stratégia célja a városi közlekedés hatásfokának javítása. A rendszer kiszámítja az irányított hálózat csomóponti forgalomirányító berendezései számára az optimális zöldidőket minden szabályozási ciklusban (90 másodperc). Az optimalizálás természetesen a forgalom aktuális állapotaitól függően történik. A szabályozási sémát a 3. ábra szemlélteti.

A stratégia a városi forgalom állapottér alakú, lineáris modelljére épül, amely gyakorlatilag a jól ismert Store and forward (Tárol és továbbít) forgalommodell (*Gazis és Potts, 1963*) hálózati szintű kibővítése. Az állapottérben leírt rendszerek elsősorú előnye, hogy közvetlenül alkalmazhatók a modern irányításelmélet hatékony technikáinak implementálására. Éppen ezért már számos forgalomirányítási stratégia született erre a

modellre építve, pl. Diakaki et al. (1999), de Oliveira et al. (2010), Tettamanti et al. (2008), Tettamanti és Varga, (2009). Mivel az állapotterez megközelítés általában komoly mátrixműveleteket von maga után, már az irányításhoz szükséges modell önmagában indukálta a MATLAB szoftver alkalmazását a rendszer szimulációjának felépítésekor.

3. ábra

Városi forgalomirányítás szabályozási sémája

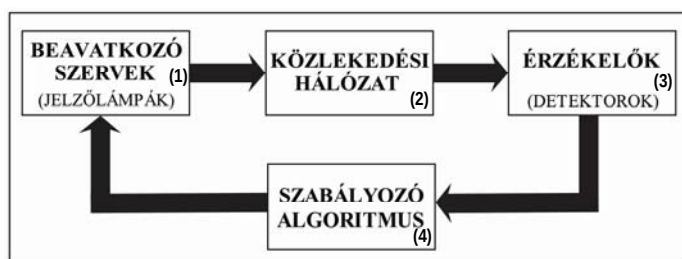


Figure 3: Control scheme of urban road traffic control

Actuators (traffic lights)(1), Traffic Network(2), Sensors (loop detectors)(3), Control Algorithm(4)

A stratégia optimalizálási motorja az ún. prediktív irányítás (García et al., 1989), amelyet számos műszaki szakterületen alkalmaznak az iparban. A MPC (Model Predictive Control) szabályozás célja ez esetben a városi hálózat útszakaszain kialakuló jármű sorhosszak minimalizálása az optimális jelzéstervek meghatározásával. A fejlesztett módszer különlegessége az adaptív képességén túl, hogy robusztus irányítást képes megvalósítani. Azaz kezelni tudja a rendszerben megjelenő állapot bizonytalanságokat (nem mérhető járműáramlás, pl. parkolás), amik megjelenése nem kiküszöbölhető nagyon jó minőségű mérőrendszer esetén sem. Amennyiben a bizonytalanságok gyakrabban, és nagyobb mértékben fellépnek, erősen megzavarhatják az online optimalizáláson alapuló forgalomirányítást. Ennek megoldására robusztus MPC módszer (Löfberg, 2003) alkalmazása szükséges, amely bizonytalanságok esetén is ki tudja számolni az optimális zöldidő-elosztást.

A szabályozó algoritmus motorja egy minimax optimalizálás, amelyet mátrix egyenlőtlenségek (LMI: Linear Matrix Inequality) segítségével felírva hatékony szemidefinit programozási feladatként (SDP: Semidefinite Programming) lehet kifejezni. Látható tehát, hogy az algoritmus leprogramozásához is nagyon hasznosan lehet felhasználni a MATLAB-ot, amely ráadásul külön LMI modullal rendelkezik ilyen számítások egyszerű megvalósításához.

A robusztus városi forgalomirányítás szimulációs keretrendszere

Az előző fejezetben részletezett fejlesztés tesztelése egy összetett feladat megvalósítását jelentette, amelyben egyaránt szükség volt a VISSIM COM, a VISSIM Signal Controller API, és a MATLAB Engine API programozás felhasználására egyaránt. A szimulációs keretrendszert tehát a 2. ábrán bemutatott séma szerint építettük fel.

A teszhálózat (4. ábra) a VISSIM grafikus felületén készült.

4. ábra

Teszthálózat robusztus forgalomirányítás vizsgálatához

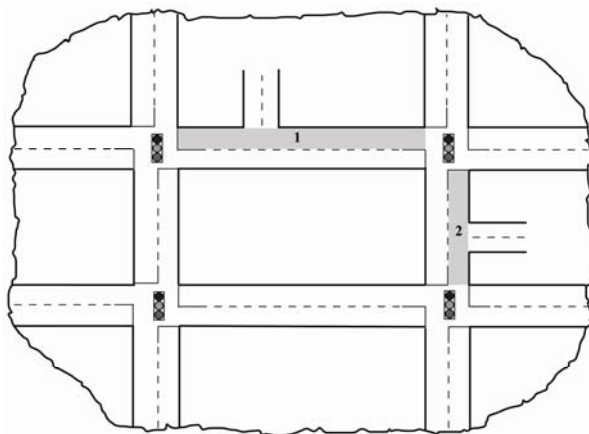


Figure 4: Test network for the analysis of robust traffic control scheme

A hálózat 4 jelzőlámpás csomópontból és 16 irányított útszakaszból áll. A szürkével jelölt szakaszok a bizonytalansággal terhelt utcákat jelentik. A bizonytalanságok dinamikáját COM felületen keresztül dinamikusan határozza meg a vezérlő C++ program. Az 1. és 2. útszakaszok aktuális állapotainak (járműszám) függvényében egy random függvény segítségével minden szabályozási ciklusban újradefiniáljuk a bizonytalanságok mértékét és irányát. A bizonytalanságot a VISSIM-ben két kisebb, nem irányított kereszteződés (4. ábra) generálja, amelyek nem mért forgalomáramlás forrásaiként, ill. nyelőiként működnek közre. A szabályozás megvalósításához a szakaszonkénti járműszámok folyamatos mérése elengedhetetlen. A VISSIM COM-on keresztül természetesen bármelyik pillanatra vonatkozóan lekérdezhető a pontos járműszám. Ugyanakkor a valóságban útszakaszonként legalább 3 detektorra van szükség ahhoz (Vigos et al., 2008), hogy releváns mérési adatokhoz jussunk. Éppen ezért a Vigos et al. (2008) által publikált eredményeket felhasználva detektorok és Kálmán-szűrő becslési eljárás (Welch és Bishop, 1995) felhasználásával valósítottuk meg a teszthálózat mérőrendszerét VISSIM Signal Controller API programozás segítségével. Maga a szabályozási algoritmus futtatása ugyanakkor MATLAB-ban történt felhasználva az LMI problémák kezelésére vonatkozó, beépített függvényeket. A teszthálózat *External* típusú forgalomirányító berendezéseit szintén a Signal Controller API segítségével valósítottuk meg. A jelészvezezőlők működési frekvenciája 1/másodperc, azaz másodpercenként frissítik a jelzőfejekre kivezért jelzéseképeket.

A szimulációs keretrendszer működése a következőképpen foglalható össze:

1. VISSIM Signal Controller API: a szabályozási ciklus (90 másodperc) végén a forgalom állapotainak becslése a detektorok mérései alapján (Kálmán-szűrő algoritmus);
2. MATLAB Engine API: mérési adatok küldése COM-on keresztül a MATLAB-nak, amely elvégzi az SDP optimalizálást;

3. VISSIM Signal Controller API: az új optimális zöldidők fogadása a MATLAB-ból, amely alapján a VISSIM-ben definiált Signal Controllerek a következő ciklus végéig vezérlik a jelzőfejeket;
4. VISSIM COM: a bizonytalan forgalomnagyságok meghatározása, és beállítása a VISSIM szimulációban a következő ciklus számára.

Szimulációs eredmények

A forgalomirányítási stratégia szimulációjának fő célja az volt, hogy a robusztus tulajdonság hatékonyságát meg lehessen vizsgálni. Az eredmények referenciájaként a nominális MPC (NMPC) szabályozást választottuk, amely kizárólag csak abban különbözik a robusztus MPC-től (RMPC), hogy nem képes a potenciálisan felmerülő a bizonytalanságok kezelésére. Az összevethetőség miatt tehát NMPC és RMPC stratégiával egyaránt lefuttattuk a szimulációkat. Három különböző szimulációs scenáriót használtunk - az érintett útszakaszok állapotának legfeljebb $\pm 10\%$, $\pm 20\%$ és $\pm 30\%$ -os arányában beállítva a bizonytalanságok mértékét. Ez azt jelenti, hogy például az első esetben, a rendszer képes maximum $\pm 10\%$ -os mértékű bizonytalanság kezelésére. Ugyanakkor a VISSIM szimulációban nem feltétlen jelent meg a $\pm 10\%$ -os érték, mivel annak mértékéről (és előjeléről) egy - zérus középértékű, normális eloszlású dinamikát megvalósító - C++ randomfüggvény döntött. Így az véletlenszerűen állt elő -10% és $+10\%$ között ciklusról-ciklusra. A állapot bizonytalanságot a VISSIM-ben nem irányított és nem mérhető csomóponti járműáramlatként modelleztük.

A szimulációs idő minden esetben 1800 másodperc volt. Az irányító algoritmus online módon minden ciklusban lefutott. Az eredményeket az 1. táblázat tartalmazza.

1. táblázat

Szimulációs eredmények

Potenciális bizonytalanság maximális értéke(1)	Forgalomtechnikai paraméterek (2)	NMPC (3)	RMPC (4)	Változás (5)
$\pm 10\%$	Járművenkénti átlagos késési idő [sec] (6)	161	159	-1.2%
	Járművenkénti megállások száma (7)	2.5	2.4	-4.0 %
	Átlagsebesség [km/h] (8)	6.8	6.9	+1.5%
	Járművenkénti átlagos utazási idő [sec] (9)	216	213	-1.4%
$\pm 20\%$	Járművenkénti átlagos késési idő [sec]	173	167	-3.5%
	Járművenkénti megállások száma	2.6	2.5	-3.8%
	Átlagsebesség [km/h]	6.4	6.6	+3.1%
	Járművenkénti átlagos utazási idő [sec]	230	223	-3.0%
$\pm 30\%$	Járművenkénti átlagos késési idő [sec]	189	173	-8.5%
	Járművenkénti megállások száma	2.9	2.6	-10.3%
	Átlagsebesség [km/h]	5.8	6.3	+8.6%
	Járművenkénti átlagos utazási idő [sec]	251	232	-7.6%

Table 1: Simulation results

Maximal potential uncertainty(1), Traffic parameters(2), NMPC(3), RMPC(4), Variation(5), Average delay time per vehicle [sec](6), Number os stops per vehicle(7), Average speed [km/h](8), Average travel time per vehicle [sec](9)

Négy különböző forgalomtechnikai paramétert vizsgáltunk a szimulációk során. Az első a *Járművenkénti átlagos késési idő*, ami azt mutatja meg, hogy mekkora az időkülönbség ahhoz az idealizált esethez viszonyítva, amikor a jármű egyedül haladna végig a hálózaton minden zavarástól mentesen. Az *Járművenkénti megállások száma*, *Átlagsebesség* és *Járművenkénti átlagos utazási idő* pedig illusztratív forgalomtechnikai jellemzők a működési hatékonyság vizsgálatára. A táblázatok Változás oszlopa a paraméterek változásának mértékét mutatja a névleges irányítási stratégiához viszonyítva.

A szimulációs eredmények egyértelmű javulás mutatnak minden paraméter tekintetében a robusztus stratégia alkalmazásánál. Ugyanakkor az is látható, hogy jelentősebb javulás a $\pm 10\%$ -os érték feletti/alatti bizonytalanság megjelenésekor jelentkezik. Összességében a szimulációs eredmények jól mutatják, hogy az adaptív jelzéstervezés robusztus tulajdonsága javította a rendszer hatásfokát az összes fontosabb forgalomtechnikai paraméter esetén.

EREDMÉNY ÉS ÉRTÉKELÉS

A cikkben bemutatott keretrendszer hasznosságát egy általunk tervezett robusztus városi forgalomirányító rendszer tesztelésén keresztül szemléltettük, amelyben a szoftverek COM-on és API-n keresztüli programozhatóságát kihasználva egy komplex, zárt hurkú szabályozási kört valósítottunk meg. A szimulációk segítségével könnyen értékelhetővé vált, hogy a tervezett stratégia milyen hatással van egy bizonytalanságokkal terhelt közlekedési hálózatra.

KÖVETKEZTETÉSEK

A bemutatott szimulációs keretrendszer hasznos eszközként szolgálhatja a felhasználókat összetettebb közúti forgalomirányítási feladatok megtervezése során. Sőt, mivel mind a VISSIM, mind a MATLAB széles körben elismert, és hatékonyan alkalmazott szoftverek, a rendszerrel gyakorlatilag teljes validáció is végezhető - megfelelő mélységű szimuláció megvalósítása esetén.

KÖSZÖNETNYILVÁNÍTÁS

A munka szakmai tartalma kapcsolódik a "Minőségorientált, összehangolt oktatási és K+F+I stratégia, valamint működési modell kidolgozása a Műegyetemen" c. projekt szakmai célkitűzéseinek megvalósításához. A szerzők köszönetet mondanak az ÚMFT TÁMOP-4.2.1/B-09/1/KMR-2010-0002 program, az NKTH és az OTKA közös OTKA CNK 78168 kutatási projekt, valamint a Bolyai János Kutatási Ösztöndíj támogatásáért.

IRODALOM

- Diakaki C., Papageorgiou M., McLean T. (1999): Application and evaluation of the integrated traffic-responsive urban corridor control strategy IN-TUC in Glasgow. Proceedings of the TRB 78th Annual Meeting, Washington D.C., USA
- García, C.E., Prett, D.M., Morari, M. (1989): Model predictive control: Theory and practice - A survey. In: Automatica, 25. 3. 335-348. p.
- Gazis, D.C. , Potts, R.B. (1963): The oversaturated intersection. Proceedings of the Second International Symposium on Traffic Theory, London, UK, 221–237. p.

- Löfberg, J. (2003): Minimax approaches to robust model predictive control. Ph.D. thesis. Linköping University, Sweden.
- MATLAB (2011): www.mathworks.com, The MathWorks, Inc, USA
- MATLAB 7 External Interfaces (2010): www.mathworks.com, The MathWorks, Inc, USA
- de Oliveira, L.B., Camponogara, E. (2010): Multi-agent model predictive control of signaling split in urban traffic networks. Transportation Research Part C: Emerging Technologies. In: Information/Communication Technologies and Travel Behaviour 18. 1. 120-139. p. ISSN 0968-090X
- Takács M. (2004): Forgalmuszimulációs elemzések egységes vizsgálati módszertanának kidolgozása, Tervezési útmutató. Gazdasági és Közlekedési Minisztérium
- Tettamanti, T., Varga, I., Kulcsár, B., Bokor, J. (2008): Model predictive control in urban traffic network management. 16th IEEE Mediterranean Conference on Control and Automation. [CD] ISBN:978-1-4244-2505-1 1538-1543. p. Ajaccio, Corsica, France
- Tettamanti T., Varga I. (2009): Elosztott közúti forgalomirányító rendszer. In: Városi Közlekedés, 49. 6. 338-341. p.
- Vigos, G., Papageorgiou, M., Wang, Y. (2008): Real-time estimation of vehicle-count within signalized links. In: Transportation Research, Part. C 16. 18-35. p.
- VISSIM (2011): www.ptvag.com, Planung Transport Verkehr AG, Germany.
- Welch, G., Bishop, G. (1995): An Introduction to the Kalman Filter. University of North Carolina at Chapel Hill. Technical Report TR95-041.
- Wiedemann R. (1974): Simulation des Straßenverkehrsflusses Schriftenreihe des Instituts für Verkehrswesen der Universität Karlsruhe, Heft 8.

Levelezési cím (*Corresponding author*):

Tettamanti Tamás

Budapesti Műszaki és Gazdaságtudományi Egyetem,
Közlekedésmérnöki és Járműmérnöki Kar, Közlekedésautomatika Tanszék
1111 Budapest, Stoczek utca 2.

*Budapest University of Technology and Economics,
Faculty of Transportation Engineering and Vehicle Engineering, Department of
Control and Transport Automation*

H-1111 Budapest, Stoczek utca 2.

Tel.: 36-1-463-2255, Fax: 36-1- 463-3087

e-mail: tettamanti@mail.bme.hu