Formális modellezés alkalmazásának lehetőségei a vasúti biztosítóberendezések területén – 2. rész

Farkas Balázs, Lukács Gábor, Dr. Bartha Tamás

Cikkünk első részében ismertettük a formális módszerek vasúti biztosítóberendezési területen történő egy lehetséges alkalmazásának hátterét és célját. Bemutattuk a vizsgálat alapját képező helyszínrajzot, a modellezési elhanyagolásokat, valamint a vágányutas és nyomvonalas biztosítóberendezési szerkesztési elvek modellezésével kapcsolatos általános megfontolásokat. Ezt követően ismertettük a két Petri hálós modellt, minek során a 2-es váltó környezetének modelljét részletekbe menően leírtuk. Végül az elkészített modellek tulajdonságait elemeztük.

Jelen írásunkban a diszkrét eseményterű rendszerek (Discrete Event Systems, DES) Petri hálós modellezésének egy alternatíváját – a nemdeterminisztikus, állapotátmeneti rendszerekkel történő modellezést – mutatjuk be. A nemdeterminisztikus átmeneti rendszerek alkalmazása a vasútbiztosítás területén kevésbé ismert, ezért röviden összefoglaljuk az erre vonatkozó elméleti hátteret, és ismertetjük a modellezés során alkalmazott eszközt (UPPAAL). Cikkünket a felhasznált eszközök összehasonlító értékelésével, valamint a modellellenőrzés folyamatának bemutatásával zárjuk.

1. Időzített automaták

Az általunk igénybe vett UPPAAL modellező és analízis szoftver az ún. „időzített automaták” formalizmusát használja. Bár modelljeinkben időzítések megadását nem alkalmaztuk (ld. 1. rész, 2.4. fejezet, 1. pont), röviden ismertetjük az időzített automaták felépítését.

Az állapotokból és a köztük definiált átmenetekből álló átmeneti rendszerek (angolul: transition systems) széles körben alkalmazott technikák a konkurens DES rendszerek modellezésére [1]. Az időzített automata egy nemdeterminisztikus átmeneti rendszer, amelyet óra változókkal egészítettek ki. Az állapotátmenetek lehetnek a rendszer által végrehajtott akciók vagy külső események által kiváltott változások. (Pl. az ismertetett modellekben esemény a vágányút kezdő- és célpontjának kiválasztása, akciók az ennek hatására végbemenő biztosítóberendezési funkciók.)

Az átmeneti rendszerekben az állapotokat egyszerűen körökkel, az átmeneteket nyilakkal (irányított élekkel) jelölik. Minden automata rendelkezik egy kezdőállapottal, melynek jele dupla kör. Egy automatában az állapotok közül mindig csak egy lehet aktív. Egy adott állapotból több engedélyezett átmenet is kivezethet, ekkor a következő állapot kiválasztása véletlenszerűen történik. Ennek korlátozására az állapotátmenetnek logikai feltételeket (ún. őrfeltételeket) is lehet szabni, melyeket általában az átmenetet jelképező élre írnak fel. Az egyszerű állapotátmeneti rendszerek a modellezés céljának megfelelően további elemekkel bővíthetők, például időzített automatákban óraváltozókkal, konkurens rendszerek leírására szinkronizációs kifejezésekkel, stb.

Az időzített automaták ún. „sűrű” időmodellt használnak, ahol egy óra változó valódi számot értékel. Minden óra szinkronban halad előre. Egy rendszert több ilyen időzített automata hálózataként, párhuzamos kompozíciójaként modelleznek. A modellt tovább bővítették az állapot részét képező, korlátozott értéktartományú diszkrét változókkal. Ezeket a változókat a programozási nyelvekhez

hasonlóan lehet használni: olvashatók, írhatók és általános aritmetikai műveletek végezhetők velük. A rendszer állapotát az automaták, az órák és a diszkrét változók értékei határozzák meg. Minden automata maga állapotot válthat vagy szinkronizálhat egy másik automatával, ami új állapotot eredményez.

Az automaták közötti szinkron kapcsolat többféle lehet. Az egyszerű eset a kézfogás elvén alapuló szinkronizáció. Ennek során a szinkronizációt kezdeményező automata jelez egy előre definiált csatornán, amire egy másik komponens reagálni képes. A szinkronizálás csak akkor lehetséges, ha mindkét automata egyszerre végre tudja hajtani a saját, adott csatornához rendelt, kezdeményező-fogadó típusú átmenetét. A másik típusú ún. üzenetszórásos szinkronizáció broadcast csatornán történik. Egyetlen küldője, de több fogadója is lehet. A küldő állapotátmenete akkor is végrehajtható, ha nincs fogadó fél.

2. Az UPPAAL eszköz

A modellezés az UPPAAL szoftver támogatásával történt. Az UPPAAL egy valós idejű rendszerek modellezésére, szimulációjára és verifikációjára használható eszköz. Kifejlesztése a svédországi Uppsalai Egyetem és a dániai Aalborgi Egyetem együttműködésében történt. Az eszköz akadémiai alkalmazásokra ingyenesen letölthető [2]. A cikk terjedelmi korlátainál fogva csak azokat a funkcióit ismertetjük, melyeket a modellezés és modellellenőrzés során felhasználtunk [3].

Egy UPPAAL modell három fő részből épül fel, a globális és lokális deklarációkból („Declarations”), az automatákból („Templates”) és a rendszerdefinícióból („System declaration”). Az UPPAAL deklarációi a C nyelvhez hasonlóan kezelik a típusokat, konstansokat, változókat és függvényeket, illetve az időzített automatáknak megfelelően a csatornákat és órákat is. Az automaták az egyes rendszerkomponensek, folyamatok modelljeit és a hozzájuk tartozó lokális deklarációkat tartalmazzák. A rendszerdefiníció a rendszert alkotó időzített automaták hálózatát adja meg. Az azonos felépítésű automaták (pl. esetünkben a váltók automatái) példányosíthatók globálisan deklarált paraméterek segítségével (objektumorientált szemlélet).

Az UPPAAL-ban az állapotok közötti átmenetekhez feltételek, akciók kapcsolódnak. Az állapotátmeneteknek négy különféle jellemzője lehet, melyek (kiértékelésük sorrendjében) a következők:

- Véletlenszerű érték sorsolása egy adott intervallumban („Select”, sárga színnel jelölt)
- Őrfeltételek vizsgálata („Guard”, zöld színnel jelölt)
- Szinkronizáció automaták között („Sync”, kék színnel jelölt)
- Értékadás („Update”, lila színnel jelölt)

A „Select” parancs hatására az állapotátmenet során megadott változóba kerül az értékkészletének egy véletlenszerűen választott eleme. Funkciója elsősorban a véletlen választás modellezése. A „Guard” őrfeltételek logikai kifejezések, melyek kiértékelése mellékhatásmentesen történik egész változókra és konstansokra, valamint ezek tömbjeinek elemeire. Az őrfeltétel szerepe, hogy a „védett” állapotátmenet csak az őrfeltételbe írt kifejezés „logikai igaz” értékre történő kiértékelése esetén mehet végbe. Az „Update” értékadások mellékhatásosak és egész változókra és konstansokra, illetve ezek tömbjeinek elemeire vonatkozhatnak, továbbá meghívhatnak függvényeket is. Feladatuk az állapotátmenetek során végrehajtott akciók modellezése.

UPPAAL-ban a szinkronizáció kétféle csatorna segítségével történhet. Egyszerű szinkronizáció esetén egy előre definiált csatornán („chan”) egy kezdeményező („Kifejezés!” formában) jelzésére egy fogadó („Kifejezés?” formában) egyszerre történő állapotátmenete mehet végbe. Az egyszerű szinkronizáció csak két fél között értelmezett, ha mindkét fél összes vonatkozó feltétele teljesül. Fogadó nélkül a kezdeményező átmenete blokkolt. (Pl. a nyomvonalas elv modelljében az automaták a kijelölést szinkronizációk segítségével adják tovább.) Az üzenetszórásos szinkronizáció broadcast csatornán („broadcast chan”) történik. Segítségével egy kezdeményező több fogadóval is szinkronizálhat, és a kezdeményező akkor sem blokkolt, ha nincs fogadásra kész másik automata. (Pl. a vágányutas elv modelljében a központi logika a feloldáshoz szinkronizációt kezdeményez a vágányútban érintett elemekkel – az oldás elmaradása a biztonság irányába való tévedés.)

Az elkészült modellek működése szimulálható az UPPAAL eszköz szimulátorában. Itt látható az automaták (már példányosított) felsorolása és az engedélyezett átmenetek listája. A szimuláció lépésenként történik, a végrehajtott átmeneteket a szimulátor tárolja.

Az UPPAAL-ba épített modellellenőrző segítségével megvizsgálható, hogy a modell teljesíti-e a formálisan jól definiált és gépileg olvasható nyelven kifejezett követelményeket. Az UPPAAL a követelmények formális leírására a TCTL (timed computation tree logic) nyelv egyszerűsített változatát használja. A TCTL nyelv útvonalakra és állapotokra vonatkozó formulákból áll. Segítségével a modellre nézve biztonsági, elérhetőségi és élőségi kritériumok teljesülése vizsgálható.

3. Modellezés UPPAAL-ban

A modellek bemutatásakor az időzített automaták és az UPPAAL eszköz rendszerelemek közötti kommunikációra (szinkronizációra) vonatkozó leíró erejét szeretnénk kiemelni, így a modellek bemutatásakor leginkább az ehhez kapcsolódó ismeretekre helyezzük a hangsúlyt. A Petri hálós modellekhez képest ez a szemlélet a rendszer leírásának egy teljesen más megközelítését jelenti. A könnyebb megértést segítendő cikkünk 1. részében bemutatott állomásrészlethez kapcsolódó biztosítóberendezési funkcionalitás modellezési tapasztalatain túl a rendszer egy elemének (a 2-es váltónak) a lényegi funkcionalitását ismertetjük.

Az UPPAAL modellek elkészítésekor ügyeltünk arra, hogy az alkalmazott jelölések igazodjanak a PetriDotNet eszközben használt jelölésekhez. A modellekben több integer (egész) illetve boolean (logikai) változót deklaráltunk. A logikai változók matematikai műveletekben is felhasználhatók (true = 1, illvetve false = 0 értékkel). A logikai változók értéke megfeleltethető a jelfogók állapotainak is: true → húzott, false → ejtett, biztonságos. A vágányút felhasználását követően, a teljes feloldás után minden változó alapállásba kerül (kivéve a váltóállást).

Cikkünkben a modellek jobb értelmezhetősége érdekében a nem releváns részek nem szerepelnek az ábrákon. Az elkészített teljes modellek (valamint cikkünk 1. része) letölthetők [4]-ről.

3.1. Vágányutas elv modellezése automatákkal az UPPAAL eszközben

A vágányutas elv átmeneti rendszer alapú modellje a vágányúti logikát (a menettervet és az elzárási tervet) megvalósító fő (vagy központi) automatából („Menet_Elzarasi”), valamint váltó („Valto”) és vágányszakasz („Vagany”) automatákból áll. A helyszínrajzon szereplő egyetlen jelző (változóként modellezve) a „Menet_Elzarasi” automata részeként került leképezésre. Ennek kapcsán megjegyezzük, hogy nagyobb modellezett állomás és több jelző esetén a jelzők külön automatába

szervezése célszerű. A figyelembe vett menet- és elzárási tervek cikkünk 1. részének 1. és 2. táblázatában láthatók. A „Menet_Elzarasi” automata struktúrája követi a Petri hálós modell struktúráját (1. rész, 4. ábra).

A vágányúti funkciók megvalósításához öt csatornatömböt (paraméterezett csatornát) definiáltunk. Ebből két csatornán a „Menet_Elzarasi” automata és a két „Valto” automata közötti állítási utasítások átadása történik. Egy-egy csatorna szolgál a váltók és a vágányszakaszok lezárásának visszajelentésére a „Valto” és „Vagany” automaták felől a „Menet_Elzarasi” automata felé. (Mind a négy csatorna egyszerű szinkronizációval megvalósított.) A lezárt vágányúti elemek feloldása egy broadcast csatornával történik, melyen szinkronizációt a „Menet_Elzarasi” automata kezdeményezhet.

A modell négy globális állandó segítségével paraméterezhető, melyek közül kettőt az elméletileg lehetséges vágányutak számának meghatározására, míg a másik kettőt az egyes automaták (váltó és vágányszakasz) példányosítására használtunk. Egyszerű, kerülővágányutaktól mentes esetben legfeljebb annyi vágányút képzelhető el, mint amennyi a start és cél elemek számának szorzata.

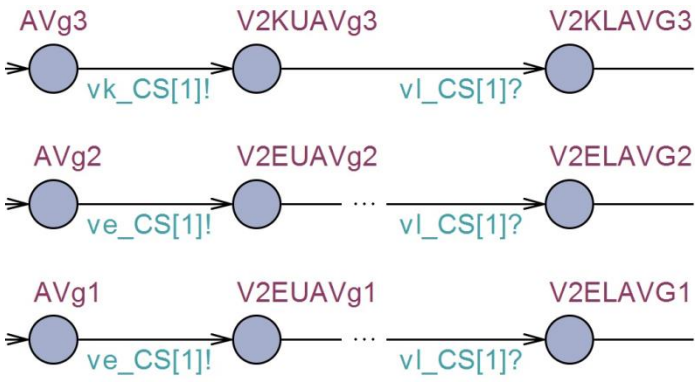
A vágányutas elv UPPAAL modelljében a vágányút beállítása egy start, majd egy cél véletlenszerű kiválasztásával kezdődik. A kiválasztott start és cél értékek szorzata adja a vágányút egyedi azonosítóját, melyet a vágányút beállítása és feloldása során használtunk fel.

A „Menet_Elzarasi” automata a menettervi függőségeket három függvény segítségével valósítja meg, melyek felépítése hasonló, mivel mindhárom a menetterv egy-egy oszlopán vagy során végez műveletet. A függvények célja a kiválasztott vágányútra vonatkozóan a menettervben megadott függőségek megvalósítása (ellenőrzése, beállítása, oldása).

A menettervi függőségek beállítása után kezdődik az elzárási terv végrehajtása, ami a Petri hálós modellhez hasonlóan (ld. 1. rész, 2.7. fejezet) történik meg. Az elzárási terv függőségeit az automata állapotaiban képeztük le. Sorrendben megtörténik a váltóállítási utasítások kiadása, a váltók átállása, majd lezáródása, illetve a vágányszakaszok lezáródása. A folyamatot követően a jelző szabadra áll.

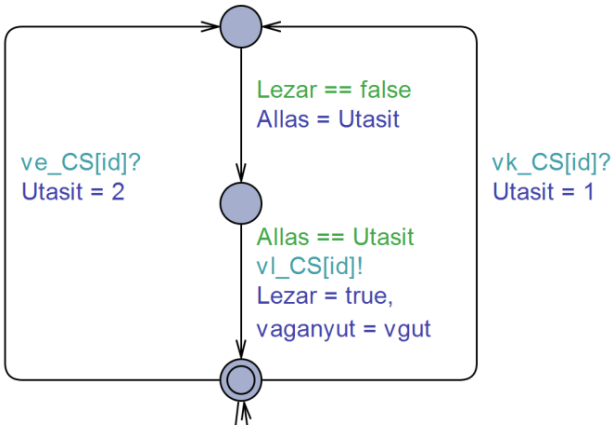
A „Menet_Elzarasi” automata a jelző „Szabad a tolatás” állásba állítása után három állapotba kerülhet, ami a három vágányhoz tartozó külön oldószakaszok működésének feleltethető meg. E működések során áll a tolatásjelző ismét továbbhaladást tiltó állásba, valamint történik a vágányúti elemek feloldása a korábban említett broadcast csatorna segítségével. Ez egyben a modellezni kívánt vágányúti funkciók végét is jelenti. A változók alapba kerülnek, és a „Menet_Elzarasi” automata új start és cél választásával új vágányút beállítására lesz képes.

A következőkben a 2-es váltóhoz tartozó funkcionalitást (kijelölés, állítás, lezárás) mutatjuk be részletesen. Először a „Menet_Elzarasi” automata a váltó vágányútban elvárt állásának megfelelő csatornán (egyenes – „ve_CS”, kitérő – vk_CS”) való szinkronizációval váltóállítási parancsot ad a 2-es váltó automatája felé (1. ábra).



1. ábra. Az elzárási terv függőségeinek végrehajtása a vágányutas elv UPPAAL modelljében

Az információt az automata eltárolja (2. ábra). Ha a váltó nincs lezárva, akkor megtörténhet az állítása. Ha a váltó állása a kiadott parancsnak megfelelő, akkor a váltó objektum automatája szinkronizál a központi logika automatájával és lezáródik.



2. ábra. A „Valto” automata állítása és lezárása a vágányutas elv UPPAAL modelljében

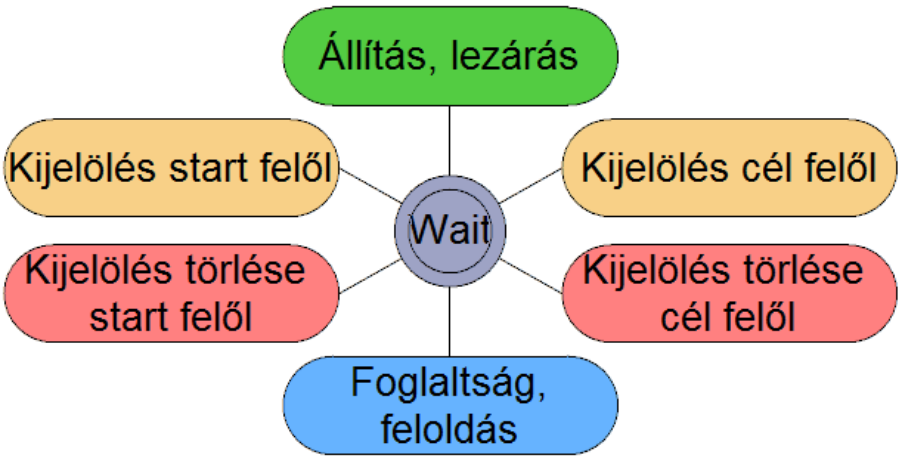
3.2. Nyomvonalas elv modellezése automatákkal az UPPAAL eszközben

A modell a kezelést megvalósító és a négy különböző funkciójú vágányúti elemet reprezentáló, példányosított automatából áll:

- Vágányút start- és célpontjának kijelölése („Kezel” automata)
- Start objektum („Start” automata)
- Váltó objektum („Valto” automata)
- Vágány objektum („Vagany” automata)
- Cél objektum („Cel” automata)

A vágányúti elemek a hozzájuk rendelt működést az egymással való kommunikáció (szinkronizáció) alapján végzik el. A négy különböző vágányúti feladathoz (kijelölés, lezárás, kijelölés törlés, feloldás) tartozó szinkronizáció négy csatornatömb segítségével valósul meg. A csatornák paraméterezését a következőképp oldottuk meg: két-két elemet egy él mindig egyértelműen köt össze (ld. 1. rész, 2. ábra). Ezért az élekhez az azonosításukra szolgáló számokat rendeltünk, amiket egy tömbben tároltuk el. Ezen tömb megfelelő celláira való hivatkozással tudnak a szomszédos elemek szinkronizációt végrehajtani.

A modell kezdőállapotában minden automata a saját kezdőállapotában („Wait” állapotban) van. Ehhez az állapothoz kapcsolódnak a szinkronizálás során végbemenő vágányúti funkciók (3. ábra). Egy-egy vágányúti funkció elvégzése után az automata a kezdőállapotába kerül vissza.



3. ábra. A nyomvonalas elvű UPPAAL modell automatáinak jellemző struktúrája

A működés a vágányút start- és célpontjának kijelölésével kezdődik, melyet a „Kezel” automata végez. Az automata „Select” paranccsal választ véletlenszerűen startot és célt. Valamennyi vágányúti működésre jellemző, hogy a „Start” és „Cel” automaták egyike indítja, vagy fordítja vissza a másik irányba. Minden automatára jellemző, hogy a különböző működésekhez kapcsolódó szinkonizálásokat csak bizonyos előfeltételek teljesülése esetén képes fogadni. A fogadott szinkronizációk során az aktuális működéshez kapcsolódó változókat beállítja, majd továbbszinkronizál.

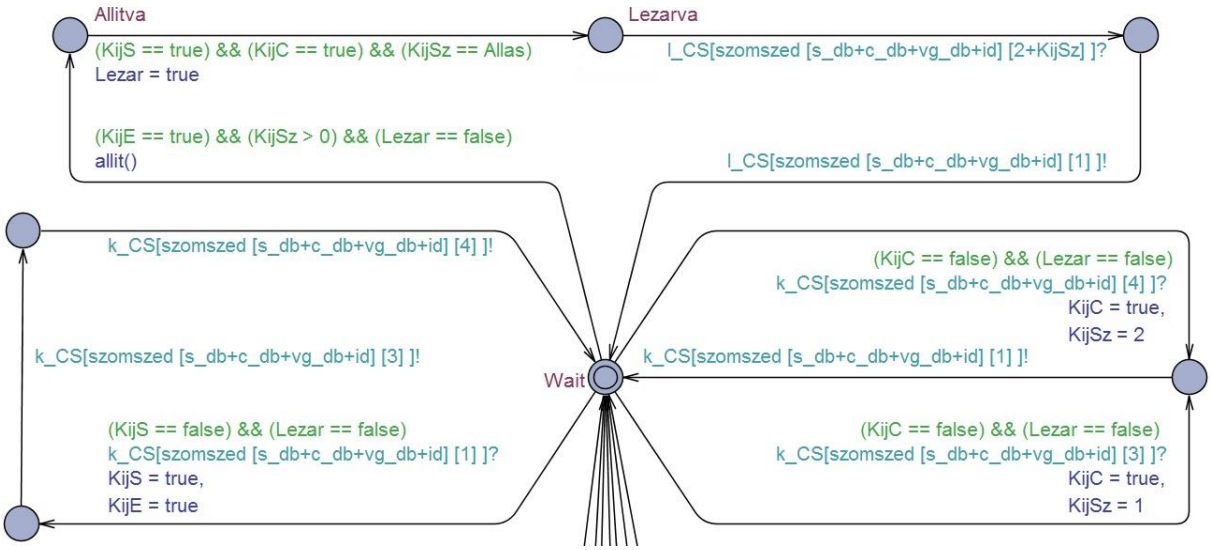
A már kijelölt és lezáródott vágányúti elemek kijelölésének törlését a „Start” automata indítja el. A kijelölés törlése a kijelöléshez képest nem külön-külön (párhuzamosan) fut végig előbb start → cél, majd cél → start irányokban, hanem minden, legalább egy irányból kijelölt elemet sorban érintve. Ha a kijelölés törlése ismét elérkezik a startba, az azt jelenti, hogy a vágányút le van zárva, és a lezárt vágányúthoz tartozó kijelölés el lett törölve.

A kijelölés törlését követően lehetőség van a „Start” automatához tartozó jelző továbbhaladást engedélyező állásba állítására. Ha minden ehhez szükséges feltétel teljesül, a jelző szabadra áll, így a menet leközlekedhet. Az UPPAAL modellben is egy megfelelően rövid (jelen példában a legrövidebb vágányúti elem hosszánál rövidebb) jármű közlekedését modelleztük. Ennek megfelelően mindig legfeljebb két szakaszt foglalhat el. A tolatómenet a váltó állásának megfelelően közlekedik tovább, tehát a szinkronizáció a váltó terelésének irányába adódik át. A folyamat során az egyes elemek felszabadulásukkal feloldónak (elemenkénti oldás).

Ezzel a vágányúti működés a végéhez ért, új vágányút beállítását szeretnénk biztosítani. Ennek érdekében a megfelelő változók inicializálásra kerülnek, és a „Kezel” automata működésével új vágányutak beállítása kezdődhet meg.

A következő részben a 2-es váltó kijelölését és lezárását ismertetjük. Az automata először fogadja a „Start” automata start felőli kijelölés szinkronizációját (4. ábra, bal oldal, vö. 3. ábra). Váltó esetén a modell megkülönböztet a start/cél felőli kijelöléseken („KijS” és KijC” változók) kívül további két kijelölési lehetőséget: a váltó eleje („KijE”), illetve szára („KijSz”) felől, amire a váltó állítása szempontjából van szükség. A „KijSz” változó három értéket vehet fel (0: nincs kijelölve, 1: bal, 2:

jobb szárán kijelölve). Az, hogy a „KijE” és „KijSz” változók közül melyik vesz fel nullánál nagyobb értéket a „KijS” vagy a „KijC” változóval egyszerre, attól függ, hogy a váltó csúcsa milyen irányban áll az adott helyszínrajzon a start és cél viszonylatában. Egy változó tartozik a váltó állásához is, ennek azonban csak két értéke lehet. (A váltó mindig csak valamely végállásában lehet, ld. 1. rész, 2.4. fejezet, 1. pont.)



4. ábra. A „Valto” automata kijelölése, állítása és lezárása a nyomvonalas elv UPPAAL modelljében

Miután megtörtént a váltó start (és egyben eleje) felőli kijelölése, az automata továbbszinkronizál a két szárán elhelyezkedő szomszédaival (III. vágány és 4-es váltó), azaz továbbadja a kijelölést. Ezt követően a váltó automata a cél felőli kijelölés szinkronizációt mindkét szárán várja azzal a feltétellel, hogy nincs cél felől kijelölve, és nincs lezárva (4. ábra, jobb oldal). Bármelyik irányból érkezzen a szinkronizálás, az a másik ágat a „KijC” változó true értékre állításával letiltja. Emellett a kijelölés irányát (melyik száron érkezett) eltárolja, majd továbbszinkronizál a start felé. A váltót lezárás előtt először a kívánt állásba kell állítani (4. ábra, felső rész). Ezt a feladatot a „Valto” automata „allit()” függvénye végzi el. Meghívásakor vizsgálja, hogy a váltó mindkét irányból kijelődött, nem foglalt és nincs lezárva. Ezek után a váltót a kijelölt szár irányába állítja. A lezáródás feltételeként a kijelölés ténye, és a szár felőli kijelöléssel egyező állás kerül ellenőrzésre. Ha ezek teljesülnek, a váltó lezáródik. Az automata ezt követően várja a kijelölt szár irányából a cél felőli lezárást ellenőrző láncot, majd továbbadja a start felé.

4. Tanulságok, tapasztalatok

A formális modellezés vasúti biztosítóberendezések területén történő alkalmazásához a feladathoz jól illeszkedő, célszerű formalizmust kell találni és a vizsgálandó rendszer modelljét úgy kell felépíteni, hogy az alkalmas legyen az ellenőrzési célkitűzések igazolására. Két részes cikkünk az előbbiekre mutat be a gyakorlati életben is jól alkalmazható példákat olyan módon, hogy két szemléletes, de elveiben eltérő formális modellező eszköz két, hasonló jellegű feladatra történő alkalmazását ismerteti. A biztosítóberendezések egyik legelterjedtebb alkalmazásai az állomási berendezések, melyek esetében a két eltérő szerkesztési elv jellemzőinek modellezésével lehetőség nyílt a különböző megközelítések vizsgálatára. Modellezett rendszernek egy olyan alkalmazást (helyszínrajz, vágányutak) választottunk, mely az elhanyagolások ellenére a legfontosabb funkciókat lefedi.

4.1. A szerkesztési elvek összehasonlítása a modellezés szempontjából

A vágányutas elv előnye, hogy csak azokhoz az elemekhez tartozik bármiféle működés egy vágányút beállítása során, melyek érintettek benne. Vágányutas elv esetén azonban egy váltó annyi vágányúttal van kapcsolatban, ahányban érintett lehet, ami nagyobb állomások és az átmenő vágányokhoz közelebb fekvő váltók esetén igen sok összeköttetést jelenthet. Ebből adódóan az egyes elemek kapcsolatainak számában is jelentős különbségek lehetnek. Ezt a nehézséget leginkább a vágányutas elv Petri hálós modelljében szemléltethetjük a vágányút feloldását végző tranzíció példáján, melyből számos él fut egyszerre végig a modell több helyére, ezáltal jelentősen csökkentve az átláthatóságot.

A nyomvonalas elv modellje bár (objektumonként) több működést tartalmaz, az egyes elemek kapcsolata mégis átláthatóbb. Összeköttetések csak „egységen belül", és egységek között vannak, így egy objektumnak mindig ugyanannyi külső kapcsolata van. A nyomvonalas elvű berendezésekben a kijelölés jóval több elemhez eljut, mint amennyi valóban érintett a vágányútban. Például a modellezett topológián a második vágányra beállított menetnél öt objektum érintett (start, 2 db váltó, vágány, cél), de további négy (2-2 db vágány és cél) kap kijelölést. Ahhoz, hogy egy elem egy vágányúthoz egyértelműen kiválasztódjon, mindkét irányból kijelölést kap. Ezek alapján látható, hogy a nyomvonalas berendezésben a vágányút-beállítás körülbelül kétszer annyi működéssel jár, mint vágányutas esetben. Emellett külön funkció szükséges a kijelölt, de a vágányútban nem érintett elemek kijelölésének megszüntetéséhez. Ennek helyes kidolgozása jelentős többlet munkaráfordítást igényelt a modellezés során.

Elmondható, hogy a modellek előállításának munkaigénye mindkét formalizmus alkalmazása során a nyomvonalas esetben a vágányutashoz képest 2-3-szoros volt. Emellett figyelembe kell venni, hogy az előbbi elv leképezéskor több hibalehetőség adódik, ami a modellek elkészítése során számos utólagos módosítást igényelhet. Azonban míg a jól megalkotott objektum modellek más helyszínrajz esetén módosítás nélkül használhatók, addig vágányutas elv alkalmazásával minden topológiára külön függőségek megvalósítása szükséges. (Ez azonban egy jól automatizálható feladat, további kutatásaink egyik iránya az automatikus transzformáció algoritmusainak kidolgozása.)

A modellezés során visszaigazoltuk azokat az alapelvekre érvényes általános megállapításokat, hogy a nyomvonalas elvű berendezéseknél minden információ lokálisan, az elemeknél kerül tárolásra. Innen kell a megfelelő információt úgy eljuttatni a vágányút valamelyik „végére", hogy minden objektum csak a szomszédjaival kommunikálhat. Ezzel szemben a vágányutas elv egy központi logikára épül, mely az egyes elemek állapotát külön-külön „felülről" lekérdezheti, ellenőrizheti.

4.2. A modellezés során használt formalizmusok és eszközök értékelése

Az egyszerű Petri hálókkal történő modellezés könnyen érthető és tanulható, mivel kevés és egyszerű működésű elem használatát szükséges elsajátítani (helyek, tranzíciók, élek, tokenek). A matematikai háttér megismerése, a kiterjesztések használata és a különböző tulajdonságok értelmezése azonban nagy ráfordítást igényel.

A Petri hálókkal szemléletesen lehet leírni különböző folyamatokat (pl. szinkronizáció). Kisebb modellek esetén egy állapot jelentése szemléletes a tokeneloszlás alapján. Nagyobb modellek, sok token esetén az átláthatóság csökken, az élek számának növekedésével a modellek nehezen értelmezhetővé válnak. Az értelmezést és bizonyos egyszerűbb hibák felfedését megkönnyítheti a szimuláció.

Az esettanulmány során a Petri hálós modellezésére igénybe vett PetriDotNet eszköz könnyen tanulható, jól kezelhető felhasználói felülettel rendelkezik. Az eszköz megfelelően kezeli az alhálókat. A modellezés és szimuláció során kihasználtuk az alhálók azon előnyét, hogy egy alhálóba szervezett rendszerelem (vagy funkciócsoport) állapottere a felhasználó számára könnyen áttekinthető és ellenőrizhető marad. Az alhálók használatakor azonban figyelemmel kell lenni arra, hogy az utólagos módosítás nehézséget okozhat.

A PetriDotNet alkalmazható a biztosítóberendezések modellezésére, mert az egyes elemek elrendezésével a vágányhálózat topológiája, vagy a táblázatoknak megfelelő mátrixos elrendezés jól lekövethető. Jól alkalmazható a vasút területén jellemző két állapottal rendelkező részrendszerek (pl.: váltóállítási utasítás vagy kijelölés +/-, jelző „Megállj!”/szabad, vágány foglalt/szabad) modellezésére (adott helyen van token, vagy nincs). További előnye, hogy a tranzíciók engedélyezettségével egyszerre több feltétel teljesülését is lehet ellenőrizni, tehát szinkronizáció egy időben egyszerre több elem között is létrejöhet.

Az UPPAAL eszköz által megvalósított automaták használatának elsajátítása a Petri hálók alkalmazásához képest több ráfordítást igényel. Az eszköz használatához a programozói ismeretek előnyt jelentenek. Az összetett szintaktika elsajátítása időbe telhet. Az eszköz súgója azonban megfelelő támogatást nyújt a kifejezések helyes használatához.

A modellezést jelentősen megkönnyíti az automaták példányosításának lehetősége. Ezáltal egy sablon megalkotása után az azonos felépítésű és viselkedésű objektumokból tetszőleges számú generálható. A globális és lokális deklarációk kezelése egyértelműen, az automata sablonokhoz rendelten történik. Az átmenetekhez tartozó kifejezések szerkesztése egyszerűen elvégezhető.

A szimuláció lehetősége az UPPAAL használata esetén is elősegíti egyszerűbb hibák felfedését. Akár a modellek készítése során is jól használható a modellellenőrző, mely ellenpéldák generálásával a hibák helyének feltárását is megkönnyíti.

Az automaták használatának előnye, hogy egy automata mindig csak egy állapotban lehet. A belőlük felépített hálózatok esetén viszont minden egyes automatának az állapotát ismerni kell. Az UPPAAL szimulációs részének használata során a változók értékei nyomon követhetőek.

Az UPPAAL eszköz is alkalmazható biztosítóberendezések modellezésére, mert az automatákkal jól leírhatók a több modulból felépülő biztosítóberendezések, és az egységek közötti kapcsolatok. A Petri hálókhoz hasonlóan logikai értékű változóikkal alkalmasak a két állapottal rendelkező részrendszerek leírására. Elsősorban vágányutas elvű berendezés modellezésénél jelent hátrányt, hogy (szinkronizációval) egyszerre csak egy objektumhoz tartozó változók értéke ellenőrizhető. Az egyszerre több elemmel kapcsolatot létesítő broadcast csatorna nem alkalmas ellenőrzésre, mert a kezdeményező fél állapotátmenetét akkor is végrehajtja, ha egyáltalán nincs, vagy az elvártnál kevesebb fogadó fél van (tehát akár az elvárt ellenőrzés teljesülése nélkül). Ezzel ellentétben például oldás modellezésére használható, mert ebben az esetben csak azok az elemek fognak feloldódni, melyek valóban készen állnak rá.

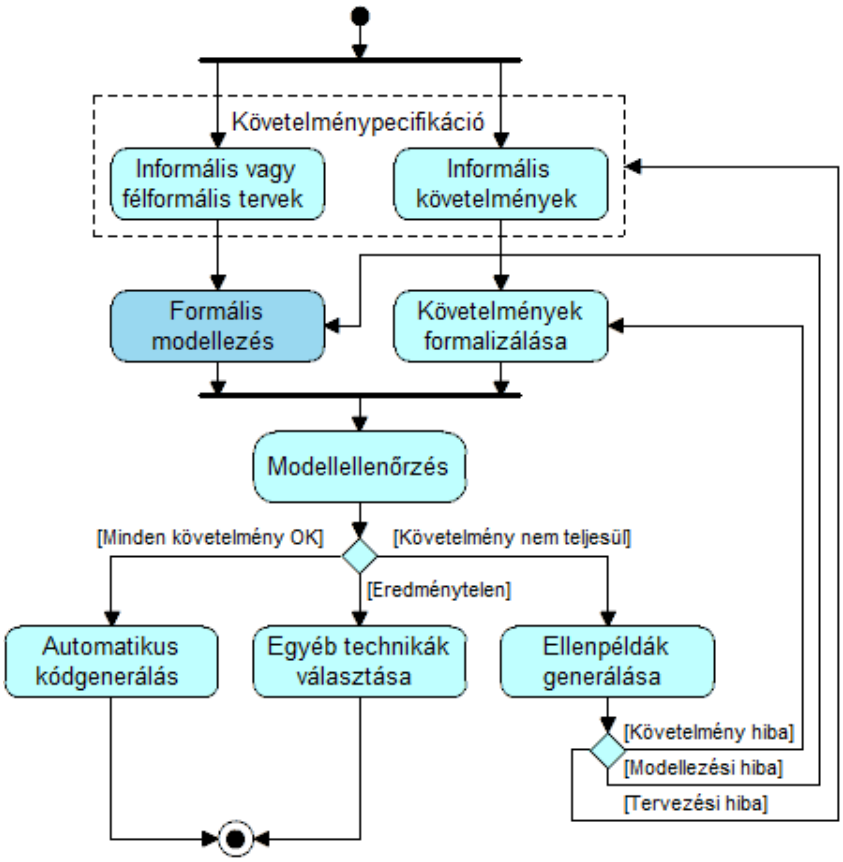
Követelmények megfogalmazására mindkét eszköz CTL-t (is) használ (ld. 2. fejezet). Azonban a két beépített modellellenőrző által elfogadott temporális logika szemantikájában, alkalmazható operátoraiban is eltér. Ezzel megnehezítik azonos követelmények azonos megfogalmazású ellenőrzését. Több modell elkészítése és egyszerre történő ellenőrzése azonban azzal az előnnyel járt, hogy a kifejezéseket az egyes modellek ellenőrzéséhez általában csak kis mértékben kellett átírni.

5. Verifikáció modellellenőrzéssel

A rendszerek komplexitásának növekedése olyan technikák alkalmazását követeli meg, amelyek támogatják és hatékonnyá teszik a mérnöki munkát, és lehetővé teszik a helyesség ellenőrzését már a tervezés korai szakaszától kezdve. A helyesség igazolásának jelentősége különösen a biztonságkritikus rendszerek esetében kiemelt fontosságú. Kétféle ellenőrzési folyamatot különböztethetünk meg: a verifikációt és a validációt. A verifikáció során egy fejlesztési életciklus fázis (vasúti területen tipikusan az EN 50126 és EN 50128 szabványok által ajánlott V-modell „leszálló" ágában szereplő fázisok) végén, a fázis elején megfogalmazott feltételek teljesülését ellenőrzik. A verifikáció ezáltal a „Helyesen építjük a rendszert?" kérdésre keresi a választ. A validáció során egy megvalósítási életciklus fázis (a V-modell „felszálló" ágában szereplő fázisok, pl. integráció, telepítés) végén azt ellenőrzik, hogy a rendszer megfelel-e a felhasználó elvárásainak. A validáció tehát a „Helyes rendszert építünk?" kérdésre keresi a választ.

A szoftver-, mind a hardverfejlesztés során egyre hangsúlyosabb szerepet kapnak az absztrakt leírások (programok, HDL nyelvek, CAD állományok), ennek megfelelően egyre szélesebb a formális módszerek alkalmazhatósága. Az elnevezés széles körben fed le különböző technikákat, mint például különböző specifikációs nyelvek és formalizmusok, automatikus tételbizonyítás, programhelyesség bizonyítás, modellellenőrzés, modelltranszformáció, de különböző absztrakciós és finomítási technikák is ide sorolhatók. Nem minden módszer alkalmazható a fejlesztés összes fázisában, egyesek inkább a tervezés, specifikáció, míg mások a verifikáció folyamatát segítik.

Az általunk elkészített modellek esetében a formális módszerek közül a modellellenőrzést használtuk fel, melynek a fejlesztési folyamatba való beágyazásának tipikus módját az 5. ábrán szemléltettük.



5. ábra. A formális módszereken alapuló fejlesztés egy lehetséges folyamatábrája (forrás: [1])

A megrendelő által elvárt, ellenőrizendő tulajdonságokat a követelményspecifikáció tartalmazza, ez a rendszerfejlesztés egyik alapvető dokumentuma. A követelményspecifikációból kétféle dokumentum keletkezik. Egyrészt elkészülnek a leendő rendszer (magas szintű, majd részletes) tervei, másrészt összegyűjtésre kerülnek a konkrét megvalósításra vonatkozó követelmények, elvárt tulajdonságok. A tervek alapján megfelelő absztrakcióval és/vagy transzformációval megalkotható a rendszer formális modellje, míg egy külön, akár párhuzamos folyamat során a szöveges követelmények formális követelményekké fogalmazhatók át. A formális modell és a formalizált követelmények képezik a modellellenőrzés bemenetét. A verifikálás során csak az bizonyítható, hogy a rendszer absztrakt leírása (terve, modellje) megfelel a követelményspecifikációnak, a megvalósított rendszer hibamentessége nem.

A modellellenőrzés egy olyan formális módszer, amely a vizsgálandó rendszer egy modelljéről és annak elvárt működését tartalmazó specifikációjáról a rendszermodell állapotai teljes halmazának (más szóval állapotterének) szisztematikus bejárásával dönti el, hogy a rendszermodell a specifikációt teljesíti-e vagy sem. Ha a bejárás specifikáció-sértést tár fel, a modellellenőrzés (eszköztől függően) ellenpéldát hoz a specifikációtól eltérő működésre [1].

Az általunk készített modellek ellenőrzéséhez [5]-ből és [6]-ból származtattunk biztosítóberendezésekre vonatkozó követelményeket. Ezek alapján a következőkben két biztonsági, funkcionális követelmény ellenőrzésére mutatunk be példát. A követelmények forrása, hogy a váltók és a jelzők között szerkezeti függések vannak oly módon, hogy:

- A váltót csak akkor lehet feloldani és átállítani, ha az őt fedező jelző „Megállj!” állásban van.
- A jelzőt csak akkor lehet „Megállj!” állásból szabad állásba állítani, ha az általa fedezett váltók helyesen állnak és ebben az állásban rögzítve vannak.

A követelményeket szétbontva, állításonként alakítottuk át először logikai, majd végül a két eszköz által feldolgozható CTL kifejezésekké. A modellellenőrzést vágányutanként végeztük. Egy biztonsági típusú követelmény formalizálására egy példát a 1. táblázatban mutatunk be. (Megjegyzés: a „ha A, akkor B” jellegű követelményeket a logikai implikáció jól ismert „¬A V B” logikai formulájába írtuk át.)

1. táblázat. Egy követelmény és annak formalizált változatai

Modell		**Kifejezés**
Követelmény		Ha a jelző szabad állásban áll, és a II. vágányra vezető vágányút van kiválasztva, akkor a 2-es és 4-es váltók megfelelő állásban állnak és le van zárva.
Petri hálós	**Vágányutas**	AG(!((Menet_Elzarasi.P_T2Sz>0)&(Menet_Elzarasi.P_LAVg2>0))|((Menet_Elzarasi.P_2E>0)&(Menet_Elzarasi.P_2L>0)&(Menet_Elzarasi.P_4E>0)&(Menet_Elzarasi.P_4L>0)))
	Nyomvonalas	AG(!((K_L_T.P_T2Sz>0)&(K_L_T.P_LVg2>0))|((K_L_T.P_2J>0)&(K_L_T.P_L2>0)&(K_L_T.P_4B>0)&(K_L_T.P_L4>0)))
UPPAAL	**Vágányutas**	A[](!((Menet_Elzarasi.T2_Szabad==1)&&(cel==2))||((Valto(1).Allas==2)&&(Valto(1).Lezar==1)&&(Valto(2).Allas==2)&&(Valto(2).Lezar==1)))
	Nyomvonalas	A[](!((Start(1).Szabad==1)&&(cel==2))||((Valto(1).Allas==2)&&(Valto(1).Lezar==1)&&(Valto(2).Allas==1)&&(Valto(2).Lezar==1)))

Az alapvető funkcionális és biztonsági követelmények ellenőrzésén túl a modelleken a megfelelő temporális logikai kifejezések megfogalmazásával egyéb tulajdonságok vizsgálata is lehetséges:

- holtpontmentesség (mindig van következő állapot, a modell sosem akad el),
- élőség (pl. ha a vonat elhaladt, előbb-utóbb minden, a vágányútban érintett elem feloldódik),
- adott visszatérő állapot (pl. a jelzőt mindig újra és újra szabadra lehet állítani, ha a megfelelő előfeltételek teljesülnek),
- stb.

A célul kijelölt tulajdonságok ellenőrzése esetében az előre meghatározott elvárt eredményt kaptuk válaszul minden esetben.

6. Összefoglalás

Kétrészes cikkünkben megvizsgáltuk a formális modellezés és modellellenőrzés vasúti biztosítóberendezésekben való alkalmazhatóságát. Az elvégzett munka alapján megállapítható, hogy mind a két formális modellezési módszer (Petri hálók illetve automaták), valamint a felhasznált PetriDotNet és UPPAAL eszközök alkalmazhatók vasúti biztosítóberendezések modellezésére és szimulációjára. Segítségükkel azonos problémák írhatók le és vizsgálhatók eltérő megközelítésekből. A modellellenőrzésben alkalmazott CTL nyelv segítségével a biztosítóberendezések különböző állapotaira vonatkozó feltételek hatásosan vizsgálhatók.

A modellezés során szem előtt tartottuk azt, hogy a modellek a későbbiekben továbbfejleszthetők legyenek, ami elsősorban az 1. rész 2.4. fejezetében részletezett elhanyagolások beépítését jelenti. Ezáltal a modellek funkcionalitásának köre bővülhet, a valóságot jobban leképező modellek kaphatók. Az alhálók és az automaták általános kidolgozásával (majdnem) bármilyen tetszőleges topológia vizsgálható.

A formális módszerek vasúti biztosítóberendezések fejlesztési folyamatában való alkalmazása jelenleg az számítástudomány és vasúti biztosítóberendezési szakterület összefogásával oldható meg hatékonyan. Kétrészes cikkünkben bemutattuk, hogy a vasúti szakterületi fejlesztőmérnökök számára a modellellenőrzés technikája egyre elérhetőbbé válik, a kidolgozott eszközkészletek rendelkezésre állnak, és viszonylag alacsony energiaráfordítással elsajátíthatók és felhasználhatók. A témával kapcsolatban további aktuális információk és a legújabb eredmények megtalálhatóak a BME KJIT tanszékén 2017-ben alapított Formális Módszerek Kutatócsoport honlapján [4].

Felhasznált irodalom

[1] ÉSIK Zoltán [et al.] *Hardver- és szoftverrendszerek verifikációja*. Szeged, Typotex, 2011., ISBN 978-963-279-497-6, URL: http://www.tankonyvtar.hu/hu/tartalom/tamop425/0008_esikgombasnemeth/Esik_Gombas_Nemeth_Hardver_1_1.html

[2] UPPAAL *Home* [számítógép-fájl]. URL: http://www.uppaal.org/ (letöltve: 2016. 11. 12.)

[3] BEHRMANN, Gerd [et al.] *A Tutorial on Uppaal 4.0* [számítógép-fájl]. Aalborg University, URL: http://www.it.uu.se/research/group/darts/uppaal/small_tutorial.pdf (letöltve: 2016. 11. 12.) pp. 1-9.

[4] FARKAS Balázs *Formális modellezés alkalmazásának lehetőségei a vasúti biztosítóberendezések területén* [diplomaterv] Budapest, BME, 2016
URL: http://www.kjit.bme.hu/index.php/hu/folap/17-kutatas/352-formalis-modszerek-kutatocsoport

[5] GAZDASÁGI ÉS KÖZLEKEDÉSI MINISZTÉRIUM *103/2003. (XII. 27.) GKM rendelet a hagyományos vasúti rendszerek kölcsönös átjárhatóságáról* [számítógép-fájl].
URL: http://net.jogtar.hu/jr/gen/hjegy_doc.cgi?docid=A0300103.GKM
(letöltve: 2016. 11. 23.)

[6] MÁV TEBK *Elektronikus állomási biztosítóberendezések feltétfüzete*. 1.02 verzió (TEBK 9240) 1996. pp. 60-61.

Német összefoglaló

Die Möglichkeiten der Anwendung von formaler Modellierung im Bereich der Eisenbahnsicherungstechnik – Teil 2

In unserem zweiteiligen Artikel erforschen wir die Anwendbarkeit der formalen Methode im Bereich der Eisenbahnen. Dafür werden die Fahrstraßen- und Spurplanstellwerksprinzipe mit zwei verschiedenen Methoden modelliert.

In dem zweiten Teil werden der Aufbau und die Funktion der entstandenen nichtdeterministischen endlichen Automaten (mit dem Programm UPPAAL) beschrieben. Am Ende des Artikels wird der Prozess der Modellprüfung kurz vorgestellt und wird eine Bewertung für die angewandten Tools gegeben.

Angol összefoglaló

Application opportunities of formal modelling in the railway interlocking systems – 2nd part

In our article we search the application opportunities of formal modeling in the railway interlocking systems. To achieve this goal we modeled the geographical and table-based principles of railway interlocking systems using two different formalisms.

In the second part of the article we summarized the experiences with the modeling of railway interlocking principles with nondeterministic automaton using UPPAAL tool. We compared and evaluated the tools which we used for modeling. We described a possible way of model checking.