

M Ű E G Y E T E M 1 7 8 2
BUDAPESTI MŰSZAKI ÉS GAZDASÁGTUDOMÁNYI EGYETEM
MÉRÉSTECHNIKA ÉS INFORMÁCIÓS RENDSZEREK TANSZÉK

**METAMODELL-ALAPÚ MODELLGENERÁLÁSI ÉS -VALIDÁLÁSI
TECHNIKÁK ÉS ALKALMAZÁSAIK**

PHD TÉZISFÜZET

SZATMÁRI ZOLTÁN

KONZULENS:
MAJZIK ISTVÁN, PHD (BME)

BUDAPEST, 2017.

1. Célkitűzések

A modellek mindig fontos szerepet játszottak a rendszer- és a szoftverfejlesztés során, többek között felhasználhatóak adatbázissémák tervezéséhez, szakterületi tudás leírásához és követelmények formalizálásához. Napjainkban még fontosabb használatuk és már a rendszertervezés számos területének központi fogalmává váltak. A modellvezérelt fejlesztés egy olyan mérnöki megközelítés, különösen a biztonság-kritikus rendszerek fejlesztése során, ahol modellek képezik a különböző fejlesztési lépések alapját mint például a követelményanalízis, rendszertervezés, implementáció és a tesztelés fázisokban.

Az Object Management Group (OMG) által kiadott Model Driven Architecture (MDA) [Obj01] specifikációval kezdődően a modell-alapú technológiák a szoftvermérnökség elemi technológiáivá váltak. Általános célú modellezési nyelvek, mint például az UML [OMG11] és a SysML [Obj12] a fejlesztési ciklus más-más szakaszaiban használhatóak, bár nem rendelkeznek pontos szemantikával. A szakterület-specifikus modellezési nyelvek egyre jobban előtérbe kerülnek, mert jobban koncentrálnak a megcélzott szakterületre és jobb eszköztámogatással és integrációs lehetőségekkel rendelkeznek. Ezen megközelítés egyik legfőbb eredménye az, hogy a szakterületi tudást egyszerűbb leírni és az adott szakterület szakértője könnyen bevonható a rendszerfejlesztés folyamatába.

1.1. Az új alkalmazási területek kihívásai

Ez a disszertáció új és feltörekvő alkalmazási területekkel foglalkozik, melyek számos új kihívást jelentenek mind a nyelvtervezés, mind pedig a modellszerkesztés, -előállítás, -validálás és -lekérdezés területén.

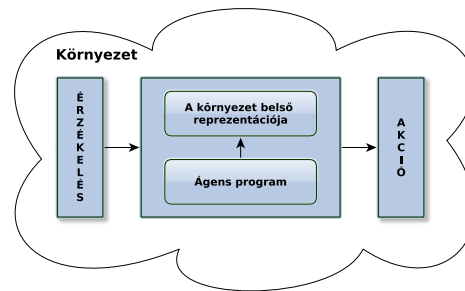
A bemutatott alkalmazási területek mind különböző ipari alkalmazásokhoz köthetőek. A velük kapcsolatban felmerülő kihívások visszavezethetőek egy közös háttérhez, a szakterület-specifikus modellezéshez, ami hasonló kérdéseket támaszt: (1) hogyan tudja a szakterületen jártos mérnök limitált informatikai háttérismeretekkel a szakterületi tudást modellezni, hogy ezzel támogassa a szakterülethez kapcsolódó problémák megoldását és (2) hogyan lehet szakterület-specifikus modelleket előállítani és validálni a fejlesztési folyamatok során?

1.1.1. Környezetfüggő autonóm rendszerek

A környezetükbe ágyazott mobil és autonóm rendszerek (AS) napjainkban különböző szakterületeken széles körben használatosak, mint például a szállítmányozási rendszerekben (pl. lézer pozicionálású targoncák), hadászati rendszerekben (pl. pilóta nélküli légi járművek), vagy a háztartásokban (autonóm porszívók). Ezt az alkalmazási területet célozta meg az R3-COP kutatási projekt, mely robosztus, biztonságos, autonóm és kooperatív robot rendszerek fejlesztésének támogatását tűzte ki céljául.

Az autonóm rendszerek (ágensek) formális definíciója [FG96] a következőképpen adható meg: *Az autonóm ágens egy olyan informatikai rendszer, mely a környezetében helyezkedik el, annak részét képezve érzékeli a környezeti változásokat és beavatkozásaival igyekszik az idő előrehaladtával a saját cselekvési tervét megvalósítani.* Az autonóm rendszer tehát egy olyan rendszer, mely emberi felügyelet nélkül, önálló döntéseket hoz és hajt végre a saját célja(i) érdekében [Con+06]. A referencia architektúrát az Ábra 1 mutatja meg. Ez a felépítés nagyon hasonló ahhoz, amit a szerzők művükben [RN03] használnak, mikor az ágens és környezetének kapcsolatát írják le.

Az ágens programja a *környezetének egy belső reprezentációját* használja, ami az ágens környezetéről alkotott saját tudását tárolja. Ennek a reprezentációnak le kell írnia minden olyan



1. ábra. Az autonóm ágens architektúráis felépítése

környezeti objektumot és eseményt, ami az ágens viselkedéséhez (vezérlő algoritmus) releváns.

Minden szoftverfejlesztési folyamatnak alapvető lépése a tesztelés. Az autonóm ágensek tesztelése egy komplex, kihívásokkal teli feladat a környezetfüggő viselkedés és a potenciális környezeti konfigurációk nagy száma miatt.

A tesztesetek készítése munka- és időigényes feladat, ezért a tesztadat-előállító folyamat megkönnyítése, vagy automatizálása kívánatos. A tesztadat-előállítási folyamat legnagyobb kihívása az ad-hoc tesztelés megelőzése és a mérhető fedettségi metrikákkal támogatott szisztematikus teszteset generálás elősegítése.

Az autonóm ágensek tesztelésének támogatásához olyan környezeteket kell előállítani, melyekben az ágens küldetése elindítható és kiértékelhető. Ezen környezetek leírhatóak modellek segítségével és szakterületi információk alapján modellgenerálási technikákkal is előállíthatóak. Ilyen szakterület-specifikus információ a metamodell, ami leírja a környezetben található elemeket valamint a jólformáltsági és szemantikus kényszerek, amik a szakterületi tudás alapján további megköteket adnak a modellekhez.

A tesztkörnyezetek leírása a környezet statikus konfigurációját tartalmazza és ezen kívül tartalmazhat dinamikus változásokat, kooperatív partnereket, emberi szereplőket, valamint konkrét üzeneteket és parancsokat tőlük. Egy ilyen generált tesztkörnyezetben az ágens környezetfüggő viselkedése megvizsgálható a funkcionális, biztonságossági és robusztussági követelmények alapján.

Ezen tesztelési megközelítés esetén az ágens „fekete doboz” tesztelése támogatott, ahol a környezet (az ágens érzékelése) szimulálható és a teszt végeredménye a szimulált teszt futtatások kimenetének elemzése alapján állítható össze.

Ezen megközelítés kihívása az, hogy (1) hogyan lehet támogatni a szakterületi szakértőt a szükséges szakterületi tudás formális modellé alakításában, és az, hogy (2) hogyan lehet biztosítani olyan modellgenerálási módszert, ami különböző tesztadat-előállítási stratégiákat és fedettségi metrikákat is kielégít.

1.1.2. Biztonságkritikus rendszerek fejlesztési folyamatai

A mindennapi életünk jelentős mértékben függ a szoftverektől, ezért a tervezési és implementációs hibák kockázatának csökkentése rendkívüli jelentőségű.

A bennmaradó szoftvertervezési hibák kockázatának csökkentése érdekében a fejlesztési folyamatokat egyre inkább szabályozzák a (szakterület-specifikus) szabványok segítségével, melyek kritériumokat fogalmaznak meg a helyes technikák és mérőszámok kiválasztására. Ezen előírások szerint, ha egy szoftver biztonság-kritikus környezetben kerül telepítésre, akkor egy

független kiértékelés szükséges annak bizonyítására, hogy a fejlesztési folyamat megfelel a releváns szabványokban foglalt kritériumoknak.

A formális modellezési technikák alkalmazása javíthatja a folyamatok kiértékelését. Ez alapvetően a szakterületi szabvány formális reprezentációját kívánja meg, ahol egy formális modell leírja a szabványban (többnyire szövegesen) megadott fogalmakat, azok relációit és a kiegészítő követelményeket.

A folyamat-tanúsítás kihívása abban rejlik, hogy a szabványban foglalt követelményeket egy szakterület-specifikus nyelv segítségével lehet leírni és ezen összegyűjtött tudás alapján egy *formális ellenőrzési módszer* kell kidolgozni. Ezt az alkalmazási területet célozza a CECRIS EU FP7 projekt [CEC13], ami módszert kíván biztosítani a kritikus rendszerek fejlesztésének, verifikálásának, validálásának és tanúsításának elősegítésére.

1.2. Az új kihívások összefoglalása

Összefoglalva, a következő *nyitott kutatási kérdéseket* azonosítottam, amelyek a munkámat motiválták és a háttérben meghúzódó általánosabb kihívások felismerése vezetett a disszertációban bemutatott tudományos eredményekhez.

1. kihívás: Szakterület-specifikus nyelvek magasszintű definiálása és validálása. Hogyan lehet szakterület-specifikus nyelveket magas szinten, felhasználóbarát módon definiálni és hogyan lehet a nyelvtervezés során a konzisztenciát ellenőrizni? Ezen kihívás céljaira az szakterületi tudás ontológiák segítségével történő leírása és a korai validálásra a kapcsolódó eszközkészlet egy ígéretes megközelítés.

2. kihívás: Környezeti modell előállításuk autonóm rendszerek teszteléséhez. Hogyan lehet a dinamikus, gyakran változó környezetben működő autonóm ágensek tesztelését támogatni tesztkörnyezeti modellek generálásával? A modellek előállításának a környezeti (meta)modellen, szakterületi megkötéseken és a tesztelési stratégiát meghatározó modell fedettségi metrikákon kell alapulnia.

3. kihívás: Modellvalidáció a fejlesztési folyamatok szabványnak való megfeleléségek ellenőrzéséhez. Hogyan lehet a biztonságkritikus rendszerek fejlesztéshez kapcsolódó, változatos szabványokban megfogalmazott különböző követelményeket formalizálni és hogyan lehet ezeket a formalizált követelményeket ellenőrizni, hogy kiértékelhető legyen a fejlesztési folyamatok szabványmegfelelése. Amennyiben a fejlesztési folyamat (a szoftverfejlesztési folyamatokra jellemző módon) egy folyamatmodellel van reprezentálva, akkor a formalizált követelmények modellvalidáció, vagy modell-lekérdezések keretében ellenőrizhetőek, lehetővé téve a szabványmegfelelési problémák és potenciális optimalizálási lehetőségek felfedését.

Ezen kutatási kérdések mögött olyan általánosabb kihívások állnak, melyek függetlenek az említett alkalmazási területtől, de kötődnek a szakterület-specifikus nyelvtervezéshez és modellezéshez. Egészen pontosan a következő általános kihívások azonosíthatóak:

1. Szakterület-specifikus nyelvek és hozzájuk kapcsolódó megkötések ontológia-alapú megtervezése és validálása,
2. Példánymodellek előállításuk metamodellek, megkötések és modellmetrikák alapján,
3. Modellek validálása modell-lekérdezésekkel leírt követelmények alapján.

Alkalmazási terület	Specifikus kutatási kérdés	Általános kihívás
Szakterületi tudás formalizálása	Szakterület-specifikus nyelvek tervezése és a hibák korai felfedezése a nyelv validálásán keresztül	Szakterület-specifikus nyelvek származtatása ontológia-alapú specifikációkból és az ontológia-alapú logikai következtetők felhasználása validálásra
Autonóm ágensek környezetfüggő viselkedésének tesztelése	Környezeti modellek létrehozása autonóm ágensek viselkedésének biztonságosságának és robusztusságának teszteléséhez	A metamodell, szakterületi megkötések és modell metrikák alapján történő modellgenerálás
Biztonságkritikus rendszerek fejlesztési folyamatainak kiértékelése	Fejlesztési folyamatok modelljeinek ellenőrzése a szabványokból származtatott követelményeknek alapján	Modell-lekérdezésekre épülő modellvalidáció

1. táblázat. Alkalmazási területek, kutatási kérdések és a kapcsolódó általános kihívások

A disszertációban javasolt technológiák épp úgy megoldást kínálnak ezen alkalmazási területeken megfogalmazott specifikus kutatási kérdésekre, mint hasonlóan feltett kérdésekre más alkalmazási területeken.

A Táblázat 1 foglalja össze a disszertációban megcélzott, a tárgyalt alkalmazási területeken felmerült kérdéseket és a kapcsolódó általános kihívásokat.

1.3. A szakterület jelenlegi helyzete

Az imént bemutatott kutatási céloknak közös a háttere: modell-alapú technikákra van igény a szakterületi tudás leírására és – az elkészített modellek alapján – modellvalidálási és -generálási technikák szükségese.

Számos megközelítés lelhető fel a modellvezérelt fejlesztéssel foglalkozó közösség körében ontológiák szakterület-specifikus nyelvek tervezéséhez és modell-alapú teszteléshez való felhasználására, de mindegyik adaptálandó, vagy kiterjesztendő ezen új alkalmazási területeken való felhasználáshoz. A következőkben összefoglalom a kutatási terület jelenlegi helyzetét és kontextusba helyezem saját munkámat.

1.3.1. Szakterület-specifikus nyelvek tervezése

A szakterület-specifikus nyelveket már számos területen sikerrel alkalmazzák rugalmasságuk és eszköztámogatásuk miatt. A szakterület-specifikus nyelvtervezés az a folyamat, amikor új szakterület-specifikus nyelv készül, opcionálisan egy létező felhasználásával, vagy teljesen a kezdetektől felépítve.

Komplex szakterület-specifikus nyelvek specifikálása különböző technikák használatát követeli meg. A szakterület-specifikus nyelv absztrakt szintaxisa általában egy metamoddellel írható le, ami kiegészíthető jólformáltsági kényszerekkel, ezáltal olyan további megkötések fogalmazhatóak meg, amit minden jólformált példánymodellnek be kell tartania. Ilyen kényszerek megfogalmazhatóak modell-lekérdezések, gráfminták (speciális modell-lekérdezés típusok) [Ber+11], vagy OCL (Object Constraint Language) nyelven megfogalmazott invariánsok segítségével. A

szakterületi követelmények általában kényszerek segítségével kerülnek formalizálásra és a követelmények példánymodellen történő teljesülése a modellellenőrzés során vizsgálható.

A szakterületi követelmények és fogalmak formalizálása az egyik legfontosabb fázisa a modellvezérelt fejlesztési folyamatnak és nyelvtervezésnek. A modellezési nyelv és a kapcsolódó követelményleírások alapján analízis, kódgenerálás és modell-alapú teszt adat előállítás is végezhető. Egy jó modellezési nyelv elkészítése egyre kevésbé bonyolult, hiszen a szakterület-specifikus nyelvtervezés már ipari gyakorlattá válik, de azért ez a továbbiakban is megkövetel informatikai és modellezési képességeket. Ezen disszertáció egyik kiemelt célkitűzése az, hogy támogatást nyújtson a nyelvtervezéshez egy olyan magasszintű megközelítés bevezetésével, ami ontológiákat és hozzájuk kapcsolódó eszközöket használ fel a szakterület-specifikus nyelvek létrehozására és validálására.

Az alacsonyabb, technológiai szinten nézve az Eclipse Modeling Framework (EMF) [EMF14] vezető szakterület-specifikus modellező technológiává vált és de facto szabvány lett a modellvezérelt fejlesztéssel foglalkozó közösség körében. Az EMF a metamodellek létrehozására egy osztálydiagramm-szerű strukturális leíró nyelvet biztosít, melynek szerkesztéséhez és a modell manipulációhoz szofisztikált eszköztámogatás áll rendelkezésre. Példánymodellek hasonlóan egyszerűen készíthetők és szerkeszthetők a grafikus (melyek GMF, GEF, vagy Graphiti technológiára épülnek), vagy szöveges (Imp, Xtext, EMFText [EMF14]) szerkesztő eszközök segítségével. A kényszerek OCL- vagy gráfminta-nyelvekkel írhatóak le és folyamatosan kiértékelhetőek a modellszerkesztés közben az esetleges hibák kiírásával és inkonzisztens elemek megjelölésével.

1.3.2. Ontológia-alapú szakterület-modellezés

A népszerű OWL2 [OWL09] és SWRL [Gli+09] nyelveket használó ontológiák már a fejlesztés nagyon korai szakaszában alkalmasak a szakterületi tudás precíz leírására. Az ontológiák természetes formalizmust kínálnak a szakterület, vagy rendszer fogalmainak felvázolására és lehetővé teszik a kiegészítő megkötések magasszintű absztrakcióval történő leírását. Számos konkrét szintaxis közül választhatunk, beleértve a gépi feldolgozásra alkalmas RDF/XML (a metamodellek funkcionális szintaxisához közeli) és a felhasználóbarát Manchester formátumot is. Mindezen túl vezetett természetes nyelvű reprezentációkat is fejlesztettek, mint például az Attempto Controlled English, vagy a FluentEditor for OWL megoldások, melyek sokat segíthetnek nyelvek leírásában, vagy a tudás dokumentálásában. Az ötletek, ahogy megszületnek, úgy inkrementálisan formalizálhatóak, köszönhetően az ontológiák *nyílt világ* feltételezésének. Az elosztott tervezés is támogatott, hiszen az egyesítés, egységesítés és konzisztencia-ellenőrzés automatikusan végrehajtható a logikai következtetők segítségével.

Úgy gondolom, hogy mind az ontológia-alapú, mind pedig a metamodell-alapú szakterület-specifikus megközelítés hasznos, hiszen a nyelvtervezés különböző szakaszában biztosítanak előnyöket. Az ontológiák magas-szintű eszközöket biztosítanak a rendszer, vagy szakterület fogalmainak és kapcsolódó megkötéseinek precíz szemantikával történő leírására és automata, meta-szintű következtetési technikákon keresztül támogatják a specifikációs hiányosságok feltárását a fejlesztés korai szakaszában, még akkor is, ha bizonyos elemek még alulspecifikáltak a modellben. A metamodell-alapú szakterület-specifikus modellezési környezetek a rendszertervezés során hatékony eszközkészletükkel támogatják a szakterületi mérnököket a példánymodell validációban és jólformáltsági kényszerek ellenőrzésében.

Az ontológiákra épülő logikai következtetők (mint például a Pellet [Cla14] vagy a RacerPro [Rac12]) a fogalmi szintű ellenőrzésre vannak optimalizálva: az ontológiák precíz szemantikájára alapozva már a tervezés korai szakaszában észreveszik az inkonzisztens specifikációkat és osztálybesorolásokat. Ezzel ellentétben, ha metamodell-alapú, OCL megkötésekkel kiegészített szakterület-specifikus eszközöket használunk, akkor nincs eszköztámogatás a metamodellben és

OCL kényszerekben található inkonzisztenciák felderítésére és még az is lehetséges, hogy egy „kielégíthetetlen” nyelv jön létre, melynek nincs érvényes példánymodellje.

A metamodell-alapú szakterület-specifikus nyelvtervezési keretrendszerek és eszközök a szakterületi mérnökök igényeire vannak szabva, így növelve a termelékenységet a funkcionálisan gazdag programozói környezetük által. Ezen eszközöket a szakterületi mérnökök hatékonyan felhasználhatják a rendszerek részletes tervezéséhez és modell manipulációkhoz. Hatékony példánymodell-ellenőrző eszközök (mint például az Eclipse OCL, vagy az EMF-INCQUERY [Ber+10]) alkalmazhatóak, melyek a tervezési szabályok, vagy megkötések megsértését gyorsan felfedezik. Ezen eszközök mindezekén túl támogatják a szakterület-specifikus alkalmazások, mint például egy modell-alapú tesztadat előállító rendszer fejlesztését.

Az ontológiák és a szakterület-specifikus nyelvek metamodelljei közötti szinergiák felfedezése és kiaknázása felkapott kutatási területté vált az elmúlt években [Hil11; WSS09; ROD10]. Egyes megközelítések [Wal+10; WSR10; WSS09] a leíró logikákra épülő szakterületi tudás specifikálásával foglalkoznak, mások [SSW09; Tao+10] példánymodell ellenőrzéssel és megint mások pedig az OWL és EMF-alapú reprezentációk közötti, csak a metamodellre szorítkozó leképezéssel. Nem létezik azonban támogatás a (1) metamodellrel ki nem fejezhető jólformáltsági kényszerek leképezésére, sem (2) az SWRL szabályok leképezésére és kiértékelésére, ahogyan (3) nem létezik egy szisztematikus benchmark, ami lefedi az elterjedt példánymodell-ellenőrző implementációkat.

2. Kutatási módszerek és új eredmények

A kutatásom kiindulópontja az a kérdés volt, hogy hogyan használjuk ki az ontológiák nyújtotta előnyöket a rendszer- és szoftverfejlesztésben, valamint a szakterület-specifikus nyelvek tervezése során.

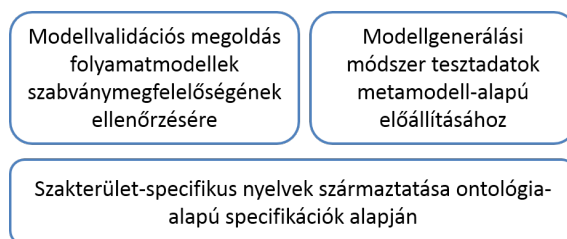
A nyelvtervezés támogatása céljából megvizsgáltam a modellvezérelt fejlesztési folyamatokban használt ontológia-alapú és a klasszikus, szakterület-specifikus nyelvek alkalmazási feltételeit. Azonosítottam a legfontosabb előnyöket és hátrányokat, majd javasoltam egy leképezést az ontológia-alapú nyelvekről a szakterület-specifikus nyelvekre.

A továbbiakban a metamodellek készítésére és a modellszerkesztés, -előállítás, -validálás és -lekérdezés területekre koncentráltam. Munkám végén adaptáltam az ontológia-alapú és modellvezérelt technológiákat a bemutatott kihívásokhoz. A modellezési technológia alkalmazási példájaként egy új, környezetet leíró modellezési megközelítést és hozzá kapcsolódó, az autonóm ágensek működésének tesztelését támogató tesztadat előállító módszert, valamint egy új, fejlesztési folyamat-kiértékelő módszert fejlesztettem ki.

Az Ábra 2 bemutatja az kutatási eredményeim egymáshoz kapcsolódását. A szakterület-specifikus nyelvek tervezéséről szóló első tézis foglalkozik az elméleti háttérismeretekkel, kapcsolódó módszerekkel és technikákkal. A másik két tézis az alkalmazási területek kapcsán tárgyalt feladatokra (tesztadat előállítás és fejlesztési folyamat-kiértékelés) javasol megoldást egy egy általános probléma megoldásán keresztül (modellelőállítás és modellvalidáció).

A tézisek bemutatásának struktúráját az Ábra 3 szemlélteti. Először bemutatom a motivációkat, majd a kihívások megoldására szolgáló módszert. A bemutatott módszer alapján javaslatot teszek a technikai megoldásra és végül az eredményeim gyakorlati alkalmazási lehetőségeit mutatom be egy esettanulmányon keresztül.

A fejezet a továbbiakban a disszertáció új tudományos eredményeit mutatja be.



2. ábra. A kutatási eredmények áttekintése



3. ábra. A tézisek struktúrája

2.1. Szakterület-specifikus nyelvek származtatása ontológia-alapú specifikációk alapján

Úgy gondolom, hogy az ontológiák (ahol a szakterületi tudást valamilyen szöveges ontológianyelv, mint például OWL2/SWRL segítségével írjuk le) és a szakterület-specifikus modellezés kombinált használata a szakterület-specifikus modellezési keretrendszerek (szakterület-specifikus nyelvek tervezését és hozzájuk kapcsolódó modellszerkesztést, -előállítást és -validálást támogató eszközök) fejlesztését felgyorsítja. Ez a tézis a szakterület-specifikus nyelvek tervezésére koncentrálna és egy ontológiákra épülő nyelvtervezési megközelítést mutat be.

Motiváció Az alkalmazási területek keretében bemutatott kihívásoknak közös háttere az, hogy mindegyik esetében szakterület-specifikus nyelvekre van szükség a szakterületi tudás leírásához és ezen nyelvek, valamint a nyelvek segítségével készített modellek validálását kell megoldani.

Ennek megfelelően, a kutatásom általános célja egy szakterület-specifikus modellezési keretrendszer fejlesztése volt az OWL/SWRL és EMF-alapú eszközök integrációjával, mely kombinálja a két megközelítés előnyeit. Megmutattam, hogy (1) a szakterület-specifikus nyelvek tervezését hatékonyan lehet támogatni szemantikus web-alapú technológiák segítségével, valamint (2) példánymodell validációs feladatok hatékonyan végrehajthatóak a szakterület-specifikus modellen, hatékony inkrementális lekérdezések formájában. Ez különösen akkor érdekes, ha az alapul szolgáló szakterületi tudás folyamatosan változik vagy fejlődik.

Módszer A tézis legfontosabb eredménye egy olyan módszer, ami biztosítja az EMF-alapú szakterületi nyelvek származtatását a szakterületi tudást leíró ontológiákból. Kezdetben megvizsgáltam a két modellezési megközelítés hasonlóságait és különbségeit és utána egy leképezést mutattam be ontológia-alapú struktúrák és az EMF-alapú szakterület-specifikus modellezési elemek között, ezzel kiterjesztve egy már meglévő, ontológiából eCore [EMF14] formátumba történő leképezést [ROD10].

Kapcsolódó publikációk [DHK07] [DLS05] alaposan tárgyalják a szakterületi tudás szöveges ontológiák segítségével történő leírását. Ez a modellezési folyamat kevesebb informatikai ismeretet kíván meg a szakterületi mérnöktől, hiszen a szakterületi tudást (fogalmakat és követelményeket) ontológiák segítségével írják le a szigorú metamodellezésre épülő technológiák helyett. Az ontológia-alapú reprezentációra építve, a kialakított struktúrák konzisztenciája már ellenőrizhető és később leképezhető EMF-alapú szakterület-specifikus modellezési nyelvekké.

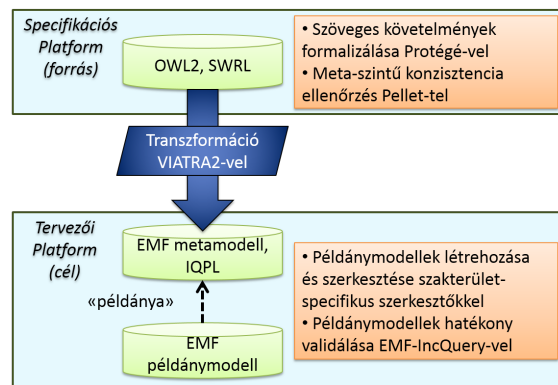
Ez a megközelítés lehetővé teszi a szakterületi mérnökök számára, hogy formalizálják és hatékonyan ellenőrizzék a követelmények konzisztenciáját már a modellezés korai szakaszában.

Technológia A technológia szintjén OWL2 és SWRL nyelvet használok a magas szintű szakterületi tudás leírására és Eclipse-alapú eszközök (pl. EMF-INCQUERY) biztosítják a szakterület-specifikus nyelv fejlesztését. Az OWL2-t és kiegészítését, az SWRL nyelvet együtt használom a szakterületi tudás leírására, míg az EMF-alapú technológiák a második fázisban már jobb eszköztámogatást biztosítanak a modellszerkesztéshez. A javasolt leképezés egy transzformációként került megvalósításra a VIATRA2 keretrendszerben [VB07].

A forrás és cél nyelvek kifejezőerejének eltérése miatt a leképezés három egymásra épülő lépésként került megvalósításra. Először az OWL2 ontológia egy részét az iparban már gyakorlatilag szabványos EMF metamodellre képezem le, mely így biztosítja az alapvető metamodell-struktúrát. A továbbiakban, a megmaradó, komplex OWL2 axiómákat képezem le egy szöveges gráfmenta nyelvre (IQPL az IncQuery gráfmenta nyelve [Ber+11]), melyeket az EMF-INCQUERY inkrementális modell-lekérdező keretrendszer hatékonyan tud kiértékelni. Végezetül, a kiegészítő SWRL szabályokat szintén gráfmentákká képeztem le.

A gráfmentákat mint a megcélzott kényszerleíró formalizmust választottam (pl. az OCL megkötések helyett), mert mind az SWRL, mind pedig az IQPL gráf-alapú nyelvek és így a két nyelv között kisebb volt a szemantikai különbség.

Az Ábra 4 áttekinti az egész transzformáció koncepcionális és technológiai felépítését.



4. ábra. Az ontológiából domain-specifikus platformra való leképezés folyamata

A megoldás különbséget tesz specifikációs és tervezési platform között.

- A specifikációs platform keretében a szakterületi mérnökök összegyűjtik a szakterületi tudást, követelményeket az SWRL-lel kiterjesztett OWL2 ontológia nyelv segítségével. Ebben a fázisban az ontológiát a Protégé [Bio14] eszközzel lehet szerkeszteni és logikai következtetők (pl. Pellet [Sir+07]) használhatóak a konzisztencia ellenőrzésére.
- A tervezési platformhoz a javasolt modelltranszformáció automatikusan származtatja az EMF metamodellt az alapvető struktúrák leírására. A komplex kényszerekhez, melyek nem fejezhetőek ki közvetlenül EMF metamodell segítségével, IQPL gráfmenták [Ber+11] készülnek.

Alkalmazás Egy modellvalidációs benchmark adaptációját mutatom be, hogy kiemeljem és mérhetővé tegyem a modellvalidáció során tapasztalható legfőbb különbségeket az ontológia-alapú eszközök és az EMF metamodellezési alapokra épülő szakterület-specifikus modellezési

eszközök között. Ez a benchmark támogatja a szakterületi mérnököt a hatékony modellreprezentáció és a modellvalidációs eszközkészlet kiválasztásában.

Szakterület-specifikus modellező eszközökhöz kapcsolódó benchmark keretrendszer már korábban lett publikálva [VSV05], mely cikkben a különböző modelltranszformációs eszközök teljesítményét vizsgálták. Kutatócsoportunk az inkrementális mintaillesztési megközelítésre koncentrált és ez alapján különböző inkrementális és nem inkrementális mintaillesztő algoritmusokat vizsgáló benchmark megközelítés került publikálásra [Ber+08]. A későbbiekben, egy inkrementális modell-lekérdezésre épülő benchmark is publikálásra került [**incquery:models**], amely szakterület-specifikus modellek szerkesztés közbeni folyamatos validációját megvalósító esettanulmányra épül. A szakterület-specifikus nyelvek felhasználásának tipikus forgatókönyveire építve a benchmark a menet közbeni modell validációt célozza meg azzal, hogy jólformáltsági kényszerek ellenőrzését vizsgálja a felhasználók által elvégzett szerkesztési lépések közben: kezdetben az egész modell kerül ellenőrzésre és a későbbiekben minden egyes modellszerkesztési lépést (pl. egy él törlését) egy azonnali újraellenőrzés követ.

Miután egy matematikailag precíz leképezést javasoltam az ontológia-alapú és az EMF-alapú nyelvdefiníciós formalizmusok között, lehetővé válik, hogy összehasonlító benchmarkot végezzünk a különböző modellvalidációs implementációk között. A korábbi inkrementális modell-lekérdezésre épülő benchmark megközelítést alapul véve, kiegészítettem a benchmark keretrendszert úgy, hogy lehetővé vált a modellvalidációs technológiák különböző modellreprezentációkon való összehasonlítása. Ezek után implementációval támogattam néhány ontológia-alapú (ABox logikai következtető) modellvalidációs eszköz teljesítményének összehasonlítását az EMF-alapú eszközökével. Mindezekon túl bevezettem egy új szintetikus szakterületi modellt („vasúti metamodel” elnevezéssel) a kapcsolódó terhelési profilokkal (lekérdezések és módosítási lépések), amely alkalmas a menet közbeni modellvalidációs esettanulmány bemutatására.

Erre a kiterjesztésre építve a szakterület-specifikus benchmark keretrendszer alkalmassá vált az elérhető validációs eszközök és különböző modellezési formalizmusok teljesítményének vizsgálatára, paraméterezhető modell-lekérdezések segítségével.

1. tézis Egy leképezést definiáltam az ontológia-alapú szakterületi fogalmak és kényszereket leíró specifikációk valamint a szakterület-specifikus metamodel és kapcsolódó gráfmenta-alapú kényszerleíró nyelvek között. Ezen leképezésre építve a szakterület-specifikus nyelvek tervezését a szakterületi tudás magasszintű ontológia-alapú reprezentációjával lehet hatékonyan támogatni.

1. OWL-alapú követelmények leképezése gráfmintákra

Egy meglévő, ontológiákról metamodellekre való leképezés kiegészítéseként definiáltam azon komplex OWL-alapú követelményleírások leképezését gráfmintákra, melyek nem fejezhetőek ki a metamodellezési formalizmussal (pl. komplex kifejezésekkel definiált osztályok). Definiáltam továbbá egy leképezést az SWRL-alapú szabályok (melyek kiterjesztik az ontológiát) és gráf minták között. Ezen gráfminták a példánymodellel értékelhetőek ki.

2. A paraméterezhető, szakterület-specifikus benchmark keretrendszer adaptálása

Kifejlesztettem egy új, szintetikus benchmark alapot (szakterületi metamodel és kényszerek formájában) és ezen benchmark alap támogatásához adaptáltam a szakterület-specifikus modellvalidációs benchmark keretrendszer implementációját. Kiterjesztettem a benchmark keretrendszert paraméterezhető modell-lekérdezésekkel, melyek biztosítják a modell-lekérdező technológiák skálázhatóságának összehasonlítását a lekérdezés komplexitás növelése mellett. Implementáltam a lekérdezéseket az ontológia-alapú eszközök összehasonlításához.

Az 1. tézis a kutatócsoportunk tagjával, Izsó Benedekkel közös kutatási terület eredményeiből építkezik. Az én kutatási eredményem az OWL és SWRL leírások leképezése a metamodell és gráfmenta formalizmusra, (2) az új benchmark alap definiálása és (3) a benchmark keretrendszer adaptálása az ontológia-alapú eszközök támogatására.

Izsó Benedek eredményei a javasolt benchmark keretrendszer kiterjesztéshez kapcsolódnak: ő implementálta a mérésre szolgáló keretrendszert, a gráfadatbázisokhoz kapcsolódó illesztéseket és végül futtatta és értékelte a méréseket.

Szárnyas Gábor folytatta ezt a kutatási irányt és adaptálta a javasolt keretrendszert elosztott környezetekhez, elkészítette az implementáció illesztését elosztott gráfadatbázisokhoz és elosztott EMF-alapú eszközökhöz (pl. IncQuery-D [Sz+14]), valamint modellmetrikákat definiált a javasolt lekérdezésmetrikákhoz hasonlóan [Sz+16].

Az 1. tézis eredményei a disszertáció 2. fejezetében foglalnak helyet. A kapcsolódó publikációk: [1], [6], [9], [10].

2.2. Modellgenerálási módszer tesztadatok metamodell-alapú előállításához

Ebben a tézisben egy modellgenerálási módszert mutatok be, ami autonóm ágensek (pl. autonóm robotok) teszteléséhez alkalmas tesztadat-előállítására használható.

Motiváció Számomra a motivációt egy olyan tesztadat előállító módszer fejlesztése nyújtotta, ami támogatja az autonóm ágensek környezetfüggő viselkedésének tesztelését a biztonságossági és robusztussági szempontok szerint. A tesztelés célja a viselkedés ellenőrzése változatos környezeti konfigurációk mellett. Ezen környezetek felépíthetőek az ágens funkcionális és biztonságossági követelményeinek leírásában szereplő egyszerűbb környezeti minták kombinálásával, vagy kiterjesztésével, mialatt a különböző környezeti objektumtípusok tesztfedettsége is vizsgálható. A tesztkörnyezetek leírhatóak példánymodellek segítségével, melyek a környezeti metamodell és tesztelési céloknak megfelelően lettek előállítva. Mindezeket túl ezen környezetek paramétereinek (különböző attribútumok értékei, az objektumok, vagy relációk száma) megfelelő megválasztásával extrém környezetek hozhatóak létre, melyek robusztusságteszteléshez használhatóak. Ilyen környezetek szisztematikus előállításához a környezet modellezése és példánymodell-generálási módszerek szükségesek.

Módszer Ontológiákra épülő környezeti modellezési megközelítést javasolok, melynek során az egyes környezeti elemek hierarchiája és kapcsolatai (objektumok és változások) precízen formalizálhatóak. Ez a környezeti metamodell közvetlenül használható lesz, amikor fontos környezeti fedettségi metrikákat definiálunk és kalkulálunk a tesztelés során.

A környezeti metamodell tartalmazhat absztrakt elemeket (fogalmakat és kapcsolatokat), ezáltal támogatva az ekvivalencia-osztályokra épülő tesztelési megoldást, azaz a tesztelés szempontjából nézve absztrakt fogalmakat és kapcsolatokat alkalmazok konkrét fogalmak és attribútumok ekvivalencia-osztályainak leírására. Ezen absztrakt relációk, értékek és osztályok bevezetése lehetővé teszi a tesztmérnökök számára a tesztelési követelmények ezen absztrakt elemekkel való definiálását a konkrét értékek helyett. Konkrétabban nézve a környezeti modellezés és teszt specifikálás szakaszban a különböző attribútumértékek alapján ekvivalenciaosztályok definiálhatóak (pl. konkrét távolsági értékek helyett bevezethető a *távoli* és *közeli* absztrakt reláció). Mindezeket túl, a környezeti metamodellre épülve a funkcionális specifikáció alapján olyan jólformáltsági és szemantikus kényszereket lehet megfogalmazni modell minták segítségével, melyek általánosságban jellemzik a szakterületet (behatárolva a valós környezeti modelleket, pl. az objektumok elrendezését, vagy az események időzítését).

A modell-alapú tesztadat-előállítás [8] a metamodell és változatos tesztelési stratégiák alapján futtatható, aminek a kimenete a generált tesztadat, ami egy olyan környezetet reprezentál, amiben az ágens tesztelhető.

A különböző tesztelési stratégiáknak megfelelően, különböző megkötések lehet megadni, ezáltal egy megkötés-halmazt létrehozva, amit az előállított tesztadatoknak ki kell elégíteniük. Ez a kényszerek alapján történő tesztadat-előállítás egy kényszerkielégítési problémaként is értelmezhető és ennek megfelelően egy kényszerkielégítési problémamegoldó alapú megközelítést javasolok a tesztadat generálásához.

A tesztadat-előállító folyamatot két fázisra osztottam, amit a kiválasztott, támogatandó teszt stratégiák, a probléma komplexitása és a megoldás hatékonysága motivált. Az első fázisban a folyamat a tesztadat strukturális megkötéseit veszi figyelembe (azaz a környezeti objektumok típusának és kapcsolatainak kiválasztása történik meg), de az absztrakciók nem kerülnek feloldásra és nem foglalozik konkrét attribútum értékekkel sem. A létrehozott tesztadat az absztrakt elemekkel még nem használható fel közvetlenül tesztelésre szimulátorban, vagy a való életben, ahol konkrét értékekre lenne szükség az autonóm ágens környezetének felépítéséhez. A második fázisban az absztrakciók kerülnek feloldásra és a korábban nem definiált attribútum értékek kerülnek meghatározásra a maradék (pl. szemantikus) megkötések és tesztelési stratégia alapján.

Összefoglalva, javasoltam egy tesztadat-előállítási módszert, ami a tesztadat-előállítási problémát egy kényszerkielégítési problémává képezi le. Ezen kényszerkielégítési problémával a megoldó program elindítható és a kimenete a megfelelő kontextusban értelmezve tesztadatot ír le, amit példánymodellként lehet ábrázolni.

Technológia A technológiai szinten a kényszerkielégítési problémák leírására a Satisfiability Modulo Theories (SMT) nyelvet választottam, míg a modellmanipulációs és -generálási feladatok megvalósítására az EMF-alapú technológiát. A javasolt leképezésnek megfelelően egy transzformáció készült, ami a tesztadat előállító problémát SMT problémává képezi le. Ez a megközelítés egy Eclipse platformra épülő tesztadat előállító keretrendszerként került implementálásra.

Alkalmazás Alkalmazási esettanulmányként megmutatom, hogy hogyan lehet a javasolt tesztadat-előállító módszert felhasználni lézer pozicionálású targoncák tesztelése során (ami az R3-COP projektben [R3C11] szereplő tesztadat előállítással kapcsolatos esettanulmányok egyike). Bemutatom a biztonságossági és robusztussági teszteléshez szükséges tesztadatok előállítása során szerzett tapasztalataimat és részletekkel szolgálok javasolt tesztadat előállító keretrendszer skálázhatósági és teljesítménybeli kérdéseivel kapcsolatban.

2. tézis Autonóm rendszerek biztonságosság és robusztusság szempontok szerinti teszteléséhez szisztematikus módszert követve fejlesztettem ki egy, a környezet leírására szolgáló modellezési megközelítést és javasoltam egy modell-alapú teszt környezet előállítási módszert különböző tesztelési stratégiák esetére.

1. Környezeti metamodell készítése

Javasoltam egy modellezési megközelítést a környezetfüggő viselkedésű autonóm ágensek környezetének leírására, ami magában foglalja a metamodell ontológia-alapú származtatását és szintaktikai (jólformáltsági) és szemantikai kényszerek definiálását.

2. A tesztadat előállító folyamat kényszerkielégítési problémaként való definiálása

Javasoltam egy kényszerkielégítési problémamegoldókra épülő tesztadat-előállító módszert. Ehhez javasoltam egy kétlépéses, visszacsatolási opcióval rendelkező modellelőállító folyamatot, mely a környezeti metamodell és a tesztadat előállítási stratégiák SMT problémára való leképezésén alapul. Megvalósítottam ezt a tesztadat-előállító módszert egy tesztadat-előállító keretrendszer formájában, validáltam autonóm járművek tesztelése során és analizáltam a skálázhatóságát és teljesítményét.

Az 2. tézis a kutatócsoportunk tagjaival, Oláh Jánossal és Micskei Zoltánnal közös kutatási terület eredményeiből építkezik. A modellgenerálás alapú tesztadat generálási folyamat közös ötletén alapulva én készítettem el az ontológia-alapú környezeti modellezési megközelítést és fejlesztettem a kétfázisú SMT-alapú tesztadat előállító módszert.

Semeráth Oszkár saját későbbi kutatásában továbbfejlesztette a tesztadat előállító keretrendszert új kényszerkielégítési probléma megoldó implementációk (pl. Alloy [Jac02]) integrálásával.

A 2. tézis eredményei a disszertáció 3. fejezetében foglalnak helyet. A kapcsolódó publikációk: [2], [8], [7] és [14].

2.3. Modellvalidációs megoldás folyamatmodellek szabványmegfelelőségének ellenőrzésére

Ez a tézis biztonságkritikus rendszerek fejlesztési (beleértve a tervezési, implementációs és verifikációs/validációs) folyamatainak kiértékelésére és tanúsítására koncentrálna.

Motiváció Ezen kutatás motivációja az volt, hogy támogassam a fejlesztési folyamatok kiértékelését egy olyan formális verifikációs módszer kidolgozásával, ami biztosítja a szakterület-specifikus szabványoknak való megfelelés automatikus ellenőrzését. Ez a kiértékelés és tanúsítása jelenleg kézzel történik, így egy automatizált megoldás sokat javíthat ezen feladat megoldásának hatékonyságán.

Módszer A biztonságkritikus rendszerek fejlesztési folyamatait megcélzó szabványok (pl. IEC61508 és EN50128) megkövetelik az alkalmazás biztonság integritási szintjének megfelelően megválasztott mérőszámok és technikák alkalmazását a fejlesztési folyamat során. Ennek megfelelően minden fejlesztési lépéshez a kötelező (M), erősen ajánlott (HR), ajánlott (R) és a nem ajánlott (NR) mérőszámok és technikák táblázatos formában elő vannak írva, ahol mindegyik technika egy eszköz, vagy eszközlánc végrehajtásával valósul meg.

A szakterület-specifikus modellezési megközelítés segítségével ezen követelmények formális reprezentációját egy szakterületi mérnök el tudja készíteni. Amennyiben a konkrét fejlesztési folyamat is folyamatmodellként rendelkezésre áll, úgy modell-alapú technikák alkalmazhatók ezen követelmények teljesülésének ellenőrzésére, azaz ellenőrizhető a fejlesztési folyamatok szabványoknak való megfelelése.

A klasszikus modellellenőrzéshez hasonló módon (ami időbeli követelmények teljesülését vizsgálja viselkedés formális modelljének elérhetőségi analízisével) itt azt vizsgáljuk, hogy a folyamatmodell kielégíti-e a szabványban foglalt követelményeket a kiválasztott mérőszámok és technikák tükrében.

Egy módszert javasoltam, ami folyamatmodellek ontológia-alapú validációját biztosítja. A szakterület-specifikus szabvány alapján elkészül a módszereket és követelményeket leíró ontológia. A fejlesztési folyamat SPEM metamodell szerinti reprezentációját ontológia-alapú folyamatmodell leírásá alakítom. Ezen reprezentáció használatával a fejlesztési folyamat szabványoknak való megfelelése már logikai következtetőkkel és lekérdező eszközökkel ellenőrizhető.

Ezen megközelítés implementációja során először a módszerek és eszközök kiválasztására vonatkozó megkövetéseket formalizáltam, majd modellezési technikákat adaptáltam, hogy leírjam a módszerek kapcsolatát (pl. hierarchiájukat, vagy a megkövetelt kombinációjukra) és az eszközök (eszközláncok) képességeit. Végezetül olyan technikát dolgoztam ki, melynek segítségével ellenőrizhető a folyamattervező mérnökök által létrehozott konkrét fejlesztési folyamat helyesége is.

A követelmények formalizálása, valamint az eszközök és módszerek modell-alapú leírása megnyitja a szabványnak megfelelő folyamatok és eszközláncok szintetizálásának lehetőségét. A folyamattervező mérnök támogatható

- hiányzó módszerek és eszközök azonosításával,
- alternatív megoldások javaslatával,
- eszközlánc minták gyűjteményének felajánlásával.

Technológia A technológiai szinten definiáltam és implementáltam egy transzformációt a SPEM-alapú folyamatmodellek és az ontológia-alapú folyamatmodellek között. Megalkottam egy Eclipse platformra és ontológia eszközökre épülő validációs keretrendszert, ami magában foglal ontológiai következtetőket (pl. Pellet) és modell-lekérdező eszközöket (pl. SPARQL lekérdező eszközök).

Alkalmazás Bemutattam, hogy a javasolt módszert hogyan lehet alkalmazni a vasúti vezérlőrendszerek szoftverfejlesztési folyamataira vonatkozó EN50128 szabvány esetén. Bemutattam továbbá a szabványban definiált módszerek alapján elkészített ontológiát és megmutattam egy fejlesztési folyamat modelljének validálását.

3. tézis Elemeztem a biztonságkritikus rendszerek fejlesztési és V&V folyamataira vonatkozó szabványokban foglalt követelményeket (kiemelten a IEC61508 és EN50128 szabványok esetében) és javasoltam formalizálásukra egy ontológia-alapú megközelítést. Ezen alapulva elkészítettem egy a fejlesztési folyamatok szabványmegfelelőségi vizsgálatára szolgáló keretrendszert.

1. *A szabványban foglalt követelmények formalizálása ontológiák segítségével*

Kifejlesztettem a fejlesztési és V&V folyamatokra vonatkozó szabványokban foglalt követelmények leírására egy ontológia-alapú módszert. Javasoltam egy eljárást ezen követelmények ellenőrzésére ontológia-alapú következtetőket és lekérdezői nyelveket felhasználva. A módszer konstruktív visszajelzést ad, támogatva a szabványnak való megfelelés elérését és az eszközök kiválasztását. Konkrét esettanulmányként formalizáltam fejlesztési folyamatokra vonatkozó követelményeket az EN50128 szabványban foglaltak alapján.

2. *SPEM-alapú folyamatmodellek ellenőrzése ontológia-alapú reprezentáció segítségével*

Egy folyamat ellenőrző megközelítést javasoltam az iparban releváns SPEM metamodell-alapú folyamatmodellek ellenőrzésére. Létrehoztam egy modellellenőrző keretrendszert ezen folyamatmodellek szabványmegfelelőségének ellenőrzésére, ami a SPEM-alapú folyamatmodellek ontológia-alapú modellekké történő leképezésére épül.

A 3. tézis eredményei a disszertáció 4. fejezetében foglalnak helyet. A kapcsolódó publikációk: [3], [4], [5], [11], [12], [13] és 15.

3. Az új tudományos eredmények alkalmazásai

Ez a fejezet a disszertáció új tudományos eredményeinek gyakorlati alkalmazásait foglalja össze.

3.1. A modellvezérelt fejlesztési rendszerek skálázhatóságának szisztematikus vizsgálata

Az 1. tézisben bemutatott összehasonlító keretrendszert adaptálta, kiterjeszti és felhasználja a Mondo (Scalable Modeling and Model Management on the Cloud) kutatási projekt. [MON15].

Ipari és akadémiai partnerek együttműködésével a kutatócsoport tagjai modell- és lekérdezőmetrikákat definiáltak, melyek segítségével analizálták a modellvezérelt technológiák és eszközök skálázhatóságát a tézisben javasolt keretrendszert felhasználva (a metamodell, a lekérdezők és a forgatókönyvek alapján).

3.2. Tesztadat előállítás autonóm ágensek teszteléséhez

A 2. tézisben definiált környezeti modellezés-megközelítés az R3-COP (Resilient Reasoning Robotic Co-operating Systems) [R3C11] kutatási projekt keretében készült és került alkalmazásra.

Ipari partnerekkel való együttműködés keretében ontológia-alapú környezeti leírások készültek, melyek alapján szakterület-specifikus nyelveket származtattam az 1. tézisben bemutatott módszer segítségével. A környezeti metamodell alapján először absztrakt tesztadatokat (teszt környezetek modelljeit) állítottam elő, majd később ezeket konkrét tesztadattá képeztem le a 2. tézisben bemutatott módszer segítségével.

3.3. Folyamatmodellek kiértékelése

A folyamatmodellek kiértékelésével kapcsolatos eredményeket (3. tézis) először a MOGENTES (Model-based Generation of Tests for Dependable Embedded Systems) EU FP7 kutatási projektben [MOG08] alkalmaztam. Ebben a projektben a fejlesztési folyamatokban lévő teszt előállítási és tesztelési fázisok szabványmegfelelőségét vizsgáltuk.

A későbbiekben a CECRIS (CErtification of CRItical Systems) kutatási projektben [CEC13] egy komplex folyamatellenőrző keretrendszer készült el, melyben kritikus rendszerekre vonatkozó teljes fejlesztési folyamatokat vizsgáltam a vonatkozó szabványoknak való megfelelés szempontjából.

4. Publikációs lista

Lektorált publikációk száma: 19
Független hivatkozások száma: 35

4.1. Tézisekhez kapcsolt publikációk

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1. tézis:	•					•			•	•					
2. tézis:		•					•	•						•	
3. tézis:			•	•	•						•	•	•		•

Folyóiratcikkek

- [1] Z. Ujhelyi, G. Bergmann, Á. Hegedüs, Á. Horváth, B. Izsó, I. Ráth, Z. Szatmári, and D. Varró. “EMF-IncQuery: An Integrated Development Environment for Live Model Queries”. In: *Science of Computer Programming* 98 (Feb. 2015). doi: 10.1016/j.scico.2014.01.004
- [2] O. Semeráth, Á. Barta, Z. Szatmári, Á. Horváth, and D. Varró. “Formal Validation of Domain-Specific Languages with Derived Features and Well-Formedness Constraints”. In: *International Journal on Software and Systems Modeling* (In press 2015)

Nemzetközi konferenciatickek

- [3] B. Polgár, I. Ráth, Z. Szatmári, and I. Majzik. “Model-based Integration, Execution and Certification of Development Tool-chains”. In: *2nd ECMDA Workshop on Model-Driven Tool and Process Integration*. 2009, p. 35
- [4] Z. Szatmári. “Standards-based Assessment of Development Toolchains in Safety-Critical Systems”. English. In: *Proceedings of the 12th European Workshop on Dependable Computing, EWDC 2009*. Ed. by H. WAESELYNCK. Toulouse, France, 2009, 4 pages. URL: <http://hal.archives-ouvertes.fr/hal-00381960>
- [5] Z. Szatmári, B. Izsó, B. Polgár, and I. Majzik. “Ontology-based assessment of software models and development processes for safety-critical systems”. In: *Monographs of System Dependability Vol. 2*. Wroclaw, June 2010, pp. 1–12. URL: <http://mycite.omikk.bme.hu/doc/88962.doc>
- [6] A. Pataricza, L. Gönczy, A. Kövi, and Z. Szatmári. “A methodology for standards-driven metamodel fusion”. In: *Proceedings of the First international conference on Model and data engineering. MEDI'11*. Obidos, Portugal: Springer-Verlag, 2011, pp. 270–277. URL: <http://dl.acm.org/citation.cfm?id=2050199.2050234>
- [7] Z. Szatmári, J. Oláh, and I. Majzik. “Ontology-based Test Data Generation using Metaheuristics.” In: *ICINCO (2)*. Ed. by J.-L. Ferrier, A. Bernard, O. Y. Gusikhin, and K. Madani. SciTePress, 2011, pp. 217–222. URL: <http://dblp.uni-trier.de/db/conf/icinco/icinco2011-2.html#Szatmari0M11>
- [8] Z. Micskei, Z. Szatmári, J. Oláh, and I. Majzik. “A Concept for Testing Robustness and Safety of the Context-Aware Behaviour of Autonomous Systems”. In: *Agent and Multi-Agent Systems. Technologies and Applications*. Ed. by G. Jezic, M. Kusek, N.-T. Nguyen, R. Howlett, and L. Jain. Vol. 7327. LNCS. Springer, June 2012, pp. 504–513. doi: 10.1007/978-3-642-30947-2_55
- [9] B. Izsó, Z. Szatmári, G. Bergmann, Á. Horváth, I. Ráth, and D. Varro. “Ontology driven design of EMF metamodels and well-formedness constraints”. In: *Proceedings of the 12th Workshop on OCL and Textual Modelling. OCL '12*. ACM. New York, NY, USA: ACM, 2012, pp. 37–42. doi: 10.1145/2428516.2428523. URL: <http://doi.acm.org/10.1145/2428516.2428523>
- [10] B. Izsó, Z. Szatmári, G. Bergmann, Á. Horváth, and I. Ráth. “Towards Precise Metrics for Predicting Graph Query Performance”. In: *28th IEEE/ACM International Conference on Automated Software Engineering (ASE 2013)*. 2013, pp. 421–431

- [11] B. Gallina and Z. Szatmári. “Ontology-based Identification of Commonalities and Variabilities among Safety Processes”. In: *16th International Conference on Product-Focused Software Process Improvement*. Dec. 2015. URL: <http://www.es.mdh.se/publications/4041->

Helyi konferenciatickek

- [12] Z. Szatmári. “Standards-based Assessment of Development Toolchains”. In: *Proceedings of the 16th PhD Minisymposium*. Budapest University of Technology, Economics, Department of Measurement, and Information Systems. 2009, pp. 20–21
- [13] Z. Szatmári. “Ontology Based Assessment of Development Processes”. In: *Proceedings of the 17th PhD Minisymposium*. Budapest University of Technology, Economics, Department of Measurement, and Information Systems. 2010, pp. 38–41
- [14] Z. Szatmári. “Supporting the Testing of Autonomous Systems Using Context Ontologies”. In: *Proceedings of the 18th PhD Minisymposium*. Budapest University of Technology, Economics, Department of Measurement, and Information Systems. 2011, pp. 34–37
- [15] Z. Szatmári. “Fejlesztési folyamatok ontológia alapú ellenőrzése”. In: *Műszaki Tudományos Füzetek - XV. FMTÜ Nemzetközi Tudományos Konferencia*. Erdélyi Múzeum-Egyesület, 2010

4.2. Tézisekhez nem kapcsolt publikációk

Folyóiratickek

- [16] I. Kocsis, Á. Tóth, Z. Szatmári, T. Dabóczi, A. Pataricza, and G. Guta. “Towards cyber-physical system technologies over Apache VCL”. in: *International Journal of Cloud Computing* 5.1-2 (2016), pp. 91–111

Nemzetközi konferenciatickek

- [17] Z. Szatmári, A. Kövi, and M. Reitenspiess. “Applying MDA approach for the SA Forum platform”. In: *Proceedings of the 2nd workshop on Middleware-application interaction: affiliated with the DisCoTec federated conferences 2008*. ACM. 2008, pp. 19–24

Helyi konferenciatickek

- [18] Z. Szatmári, Z. Micskei, and I. Majzik. “Monitor komponensek okostelefon platformra”. In: *Műszaki Tudományos Füzetek - XIX. FMTÜ Nemzetközi Tudományos Konferencia*. Erdélyi Múzeum-Egyesület, 2014
- [19] I. Kocsis, A. Pataricza, G. Huszerl, B. Izsó, Z. Szatmári, Á. Tóth, D. Varró, and A. Voros. “Informatikaoktatás felhőben: egy új oktatási modell bevezetése”. In: *BeleSTEM - Felsőoktatási jó gyakorlatok a tudomány, a technológia, a műszaki tudományok és a matematika szolgálatában*. Tempus Közalapítvány, 2014, pp. 36–41

Diplomamunka

- [20] Z. Szatmári. “Applying MDA approach for the SA Forum platform”. Master’s thesis. Budapest University of Technology and Economics, 2008

Hivatkozások

- [Ber+08] G. Bergmann, Á. Horváth, I. Ráth, and D. Varró. “A Benchmark Evaluation of Incremental Pattern Matching in Graph Transformation”. In: *Proc. 4th International Conference on Graph Transformations, ICGT 2008*. Vol. 5214. Lecture Notes in Computer Science. Acceptance rate: 40%. Springer. 2008, pp. 396–410.
- [Ber+10] G. Bergmann, Á. Horváth, I. Ráth, D. Varró, A. Balogh, Z. Balogh, and A. Ökrös. “Incremental Evaluation of Model Queries over EMF Models”. In: *Model Driven Engineering Languages and Systems*. Vol. 6394. Lecture Notes in Computer Science. 2010, pp. 76–90. URL: http://dx.doi.org/10.1007/978-3-642-16145-2%5C_6.
- [Ber+11] G. Bergmann, Z. Ujhelyi, I. Ráth, and D. Varró. “A Graph Query Language for EMF models”. In: *Theory and Practice of Model Transformations, ICMT 2011*. Vol. 6707. LNCS. Springer. 2011, pp. 167–182. DOI: 10.1007/978-3-642-21732-6_12.
- [Bio14] S. C. for Biomedical Informatics Research. *Protégé ontology editor*. <http://protege.stanford.edu/>. 2014.
- [CEC13] CECRIS. *CErtification of CRITICAL Systems*. EU FP7 research project. 2013. URL: <http://www.cecris-project.eu/>.
- [Cla14] Clarkparsia. *Pellet: OWL 2 Reasoner for Java*. <http://clarkparsia.com/pellet/>. 2014.
- [Con+06] J. Connelly, W. Hong, R. Mahoney Jr, and D. Sparrow. “Current challenges in autonomous vehicle development”. In: *Defense and Security Symposium*. International Society for Optics and Photonics. 2006, pp. 62300D–62300D.
- [DHK07] G. Dobson, S. Hall, and G. Kotonya. “A Domain-Independent Ontology for Non-Functional Requirements”. In: 2007. DOI: 10.1109/ICEBE.2007.76.
- [DLS05] G. Dobson, R. Lock, and I. Sommerville. “Quality of service requirements specification using an ontology”. In: 2005.
- [EMF14] EMF - The Eclipse Project. *Eclipse Modeling Framework*. <http://www.eclipse.org/emf/>. 2014.
- [FG96] S. Franklin and A. Graesser. “Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents”. In: *Proc. of the Third International Workshop on Agent Theories, Architectures, and Languages*. 1996.
- [Gli+09] B. Glimm, M. Horridge, B. Parsia, and P. F. Patel-Schneider. *A Syntax for Rules in OWL 2*. 2009.
- [Hil11] G. Hillairet. *EMFTriple: a tool that brings semantic web languages to the EMF*. <http://code.google.com/a/eclipseorg/p/emftriple/>. 2011.
- [Jac02] D. Jackson. “Alloy: a lightweight object modelling notation”. In: *ACM Trans. Softw. Eng. Methodol.* 11.2 (2002), pp. 256–290. DOI: <http://doi.acm.org/10.1145/505145.505149>.
- [MOG08] MOGENTES. *Model-based Generation of Tests for Dependable Embedded Systems*. EU FP7 research project ID: ICT-216679. 2008. URL: <http://mogentes.eu/>.
- [MON15] MONDO. *Scalable Modeling and Model Management on the Cloud*. EU FP7 research project. 2015. URL: <http://www.mondo-project.org/>.
- [Obj01] Object Management Group. *Model Driven Architecture — A Technical Perspective*. <http://www.omg.org>. 2001.

- [Obj12] Object Management Group. *SysML v1.3 Specification*. <http://www.omg.org>. 2012.
- [OMG11] Object Management Group. *Unified Modeling Language (UML) 2.4.1 Superstructure Specification*. formal/2011-08-06. 2011.
- [OWL09] OWL Working Group. *OWL 2 Web Ontology Language*. <http://www.w3.org/2007/OWL/>. 2009.
- [R3C11] R3-COP. *Resilient Reasoning Robotic Co-operating Systems*. ARTEMIS research project nr. 100233. 2011. URL: <http://www.r3-cop.eu/>.
- [Rac12] Racer Systems GmbH. *RacerPro*. <http://www.racer-systems.com/products/racerpro/>. 2012.
- [RN03] S. Russell and P. Norvig. *Artificial Intelligence. A Modern Approach*. Second. Pearson Education Inc., 2003.
- [ROD10] T. Rahmani, D. Oberle, and M. Dahms. “An Adjustable Transformation from OWL to Ecore”. In: 2010. URL: http://dx.doi.org/10.1007/978-3-642-16129-2%5C_18.
- [Sir+07] E. Sirin, B. Parsia, B. Grau, A. Kalyanpur, and Y. Katz. “Pellet: A Practical OWL-DL Reasoner”. In: *Web Semantics: science, services and agents on the World Wide Web 5.2* (2007), pp. 51–53.
- [SSW09] E. Sirin, M. Smith, and E. Wallace. “Opening, Closing Worlds - On Integrity Constraints.” In: *OWLED*. Vol. 432. CEUR Workshop Proceedings. 15, 2009. URL: <http://dblp.uni-trier.de/db/conf/owlled/owlled2008.html#SirinSW08>.
- [Szá+14] G. Szárnyas, B. Izsó, I. Ráth, D. Harmath, G. Bergmann, and D. Varró. “IncQuery-D: A Distributed Incremental Model Query Framework in the Cloud”. In: *ACM/IEEE 17th International Conference on Model Driven Engineering Languages and Systems, MODELS 2014*. Acceptance rate: 26%. Springer. 2014.
- [Szá+16] G. Szárnyas, Z. Kővári, Á. Salánki, and D. Varró. “Towards the Characterization of Realistic Models: Evaluation of Multidisciplinary Graph Metrics”. In: *ACM/IEEE 19th International Conference on Model Driven Engineering Languages and Systems, MODELS 2016*. 2016.
- [Tao+10] J. Tao, E. Sirin, J. Bao, and D. L. McGuinness. “Integrity Constraints in OWL”. In: *AAAI*. 2010.
- [VB07] D. Varró and A. Balogh. “The model transformation language of the VIATRA2 framework”. In: *Science of Computer Programming 68.3* (2007), pp. 214–234.
- [VSV05] G. Varro, A. Schurr, and D. Varro. “Benchmarking for graph transformation”. In: *Visual Languages and Human-Centric Computing, 2005 IEEE Symposium on*. IEEE. 2005, pp. 79–88.
- [Wal+10] T. Walter, F. Silva Parreiras, S. Staab, and J. Ebert. “Joint Language and Domain Engineering”. In: *Proc. of 6th European Conference on Modelling Foundations and Applications, ECMFA 2010, Paris*. Vol. 6138. LNCS. 2010.
- [WSR10] T. Walter, H. Schwarz, and Y. Ren. “Establishing a Bridge from Graph-based Modeling Languages to Ontology Languages”. In: *Proceedings of the of the Third Workshop on Transforming and Weaving Ontologies in Model Driven Engineering (TWOMDE) at TOOLS*. 2010.

- [WSS09] T. Walter, F. Silva Parreiras, and S. Staab. “OntoDSL: An Ontology-Based Framework for Domain-Specific Languages”. In: *ACM/IEEE 12th International Conference on Model Driven Engineering Languages and Systems, 12th International Conference, MODELS 2009*. Vol. 5795. LNCS. 2009, pp. 408–422.