



Budapesti Műszaki és Gazdaságtudományi Egyetem
Méréstechnika és Információs Rendszerek Tanszék

Az LTSmin modellellenőrző vizsgálata és kiterjesztése

Hegedüs Levente, BSc mérnökinformatikus hallgató

Konzulens: Molnár Vince, MIT

Rendszertervezés ágazat

Önálló laboratórium összefoglaló

2015/16. II. félév

Feladatom az LTSmin nevű modellellenőrző eszköz vizsgálata volt, melyet a Twente-i Egyetemen fejlesztenek. Ennek a programkönyvtárnak az előnye, hogy többféle ellenőrző algoritmust tartalmaz, és az interfészén, PINS-en (partitioned next-state interface) keresztül sokféle modell-leíró megoldást tud kezelni, az ellenőrző tetszőleges algoritmusával.

A feladatra a motivációt az adta, hogy a Hibatűrő Rendszerek Kutatócsoportjánál fejlesztett PetriDotnet nevű modellellenőrző programhoz is hasznos lenne egy ilyen interfész, mely különböző modelleket tud kezelni. A hosszú távú cél a PINS és a modellellenőrző közé készítendő wrapper, mellyel a PINS-et össze lehetne kötni a PetriDotnettel.

A feladat nehézségét az adja, hogy az LTSmin-t és a PINS-et C-ben írták, Linux operációs rendszerre, míg a PetriDotnet .NET platformon fut, C# nyelven megírva. A másik probléma a Twente-iek munkájának szinte teljes dokumentálatlansága volt.

Ezért a feladatom lényege a PINS megismeréséről szólt, melyet az interfész modellekhez kapcsolódó részével végeztem. Ennek során (és már a feladat megismerésekor is), nyilvánvalóvá vált, hogy nem csak a PINS ellenőrzőhöz kapcsolódó, de a modellek felőli oldalához is érdemes .NET wrappert készíteni, bár ez a PetriDotnet project számára irreleváns, viszont az LTSmin-nek hasznos lenne, hiszen így modell-leírások készülhetnének hozzá .NET platformon.

A konkrét feladat, amit elvégeztem azzal kezdődött, hogy lehetővé tettem az egyébként Linuxra írt program futtatását Windowson, és ezáltal lehetőség nyílt rá, hogy VisualStudioval használható legyen.

Ezután készítettem egy Petri hálót szimuláló modult C++-ban, mely fájlból beolvassa képes felépíteni a szükséges adatszerkezetet tetszőleges háléhoz, és abban tranzíciókat tüzelni. Ezt egy egyszerű konzolos alkalmazással teszteltem.

Munkám következő része az volt, hogy az így elkészült modell-leírást a PINS Dlopen-API nevű modell interfészén keresztül használjam az LTSmin-nel. Ennél az interfésznél a modellt egy DLL-ben kell implementálni, melyet az ellenőrző tölt be.

Mivel a cél az volt, hogy a PINS-et objektum orientált környezetben használjuk, ezért a Twente-i Egyetem egyik példa modelljéhez (Sokoban példa) is készítettem egy osztályt, mely ugyanazt a C++-ban általam készített interfészt implementálja, mint a Petri hálós modell, ezzel jól demonstrálja a megfelelő objektumorientáltságot.

A kapott feladataimat sikerült elvégezni. Ha nem is teljesen, de megismerkedtem a PINS interfésszel, és felhasználtam VisualStudióban, objektum orientált környezetben, de egyelőre még csak natív C++ kódban.

Az így elkészült munka viszont már jó alapot ad a további folytatásra, hogy a PINS .NET környezetben is működhessen.