

Magasszintű hibamodellek validációja hibainjektálással

Pintér Gergely, műszaki informatika szak V. évf.

Konzulens: Dr. Pataricza András, BME MIT

Ahogy napjainkban a számítástechnikai rendszerek egyre nagyobbak és összetettebbek lesznek egyre magasabb megbízhatósági követelményeknek kell megfelelniük. Ezen kívánalmak teljesítése érdekében a termékekbe hibatűrési mechanizmusokat építenek. Hibatűrő rendszerek fejlesztése során filozófiai probléma, hogy a hibatűrési módszereket a fejlesztés korai szakaszában meg kell tervezni és hatékonyságukra becslést kell adni, a hibák viszont lényegében csak a megvalósítás után derülnek ki.

Ennek szokásos megoldási módja az, hogy a rendszer magasszintű specifikációjában hibákat tételeznek fel, és az így kapott modelleket alacsony szinten ellenőrzik abból a szempontból, hogy a valóságos hibák mindegyike leírható-e magas szinten, illetve, hogy nem túl redundáns-e a hibamodell, nem próbál-e nem létező hibák ellen védekezni. Néhány berendezésen benchmark szerű vizsgálatokat végeznek, ezek valósághűségét fizikai szinten ellenőrzik és amennyiben a magas szinten leírt hibamodell a tényleges hibák nagy részét leírja egyéb konstrukciónál is ilyen hibákat tételeznek fel.

Az irodalomban számos példa olvasható fizikai hiba és logikai hibahatás szintű hibamodellezésre és a hibamodellek validációjára. Az itt bemutatott megközelítés a mára ipari szabvánnyá vált UML modellező nyelven leírt rendszerekkel foglalkozik. Alapgondolata, hogy az UML-ben modellezett rendszer hibamodelljét automatikusan, az UML metamodell szintjén generáljuk, majd kódgenerálás után a hibamodellt alacsony szinten hibainjektálással validáljuk és egy visszacsatolási mechanizmussal finomítjuk. A kutatás célja megvizsgálni, hogy a más absztrakciós szinteken már létező automatikus hibamodell generálási módszereknek (pl. VHDL funkcionális hibamodellek) van-e megfelelőjük a legmagasabb szoftvermodellezési szinten.

A kutatás támogatására egy tesztelő környezet készült, amely segítséget nyújt a munkához az UML modell feldolgozásától a hibamodell értékeléséig. A programcsomag tartalmaz egy kódgenerátort, egy hibainjektort, egy watchdog alkalmazást és néhány kisebb eszközt a hibamodell értékelésére, illetve egyszerű modellek automatikus generálására.

A dolgozat bemutatja az elméleti háttérrel, tartalmazza a programok dokumentációját és egy esettanulmányt, amely egy benchmark alkalmazás példáján vázolja, hogyan működik a tesztkörnyezet, és elemzi a generált hibamodell jóságát, gyengeségeit.