Nagy megbízhatóságú elektronikus közlekedési alrendszerek RAMS paramétereinek kezelése

Ph.D értekezés

Szabó Géza
okleveles villamosmérnök

témavezető:
Dr. Bokor József
tanszékvezető egyetemi tanár

BME Közlekedésautomatikai Tanszék

Budapest, 2008

Alulírott Szabó Géza kijelentem, hogy ezt a doktori értekezést magam készítettem és abban csak a megadott forrásokat használtam fel. Minden olyan részt, amelyet szó szerint, vagy azonos tartalomban, de átfogalmazva más forrásból átvettem, egyértelműen, a forrás megadásával megjelöltem.

Budapest, 2008. november 24.

Aláírás

Tartalomjegyzék

1. ELŐSZÓ **1**
1.1 Igény a nagy megbízhatóságra 1
1.2 Motivációs háttér 2
1.3 A disszertáció felépítése 2
1.4 Köszönetnyilvánítás 3

2. BEVEZETÉS **5**

3. A MEGBÍZHATÓSÁGI SZINT DEFINIÁLÁSA ÉS VIZSGÁLATA **7**
3.1 Definíciók 7
3.2 A biztonsági (RAMS) paraméterek specifikálása és kezelése 10
3.3 Analízis eljárások 12
3.4 Becslési eljárások 13
3.4.1 Bevezetés 13
3.4.2 Alkatrész számlálás módszere 13
3.4.3 Laboratóriumi információk figyelembevétele 15
3.4.4 Tapasztalati adatok figyelembevétele 15
3.5 Ellenőrzési listák és kockázati indexek 15
3.6 Meghibásodási módok és hatások analízise 16
3.6.1 FMEA jellemzői 16
3.6.2 FMEA módszertan 17
3.6.3 Kritikusság vizsgálata 18
3.7 Az eseményfa analízis 19
3.8 A hibafa analízis 20
3.8.1 Bevezetés 20
3.8.2 A hibafa-analízis módszertana 21
3.8.3 A hibafa-analízis matematikai háttere 23
3.9 A Markov analízis (Automata kockázati analízis) 27

4. A HIBA-ADAPTÍV FUNKCIÓK ÉS ANALÍZISÜK **28**
4.1 Bevezetés 28
4.2 Hiba-aktív és hiba-adaptív funkciók 29
4.3 Az adaptivitás kezelése a hibafa-analízisben 30
4.3.1 Makro-modellek alkalmazása 30
4.3.2 Markov-lánccal történő modellezés 32
4.3.3 Többállapotú hibafa-analízis alkalmazása 33
4.3.4 Zárt formulák alapján történő feldolgozás 34

5. A MEGBÍZHATÓSÁG IDŐFÜGGÉSE **37**
5.1 Bevezetés 37
5.2 A meghibásodási ráta időbeli függése 39
5.3 A rendelkezésre állás időfüggése 40
5.3.1 Bevezetés 40
5.3.2 Konstans rendelkezésre nem állású komponens 40
5.3.3 Nem javítható komponens 41

5.3.4 Folyamatosan ellenőrzött, javítható komponens 42
5.3.5 Az időfüggvény módosítása megfigyelési adatok alapján 43
5.3.6 Periodikusan tesztelt komponens 44
5.4 Komplex modellek 47
5.4.1 Bevezetés 47
5.4.2 Folyamatosan figyelt, nem javítható komponens modell 47
5.4.3 Periodikusan tesztelt, nem javítható komponens modell 47
5.4.4 Periodikusan tesztelt, javítható komponens modell 48

6. AUTOMATIKUS MODELLGENERÁLÁS 50
6.1 Bevezetés 50
6.2 Kapcsolódó munkák 50
6.3 Hardver és funkcionális rendszerleírás 51
6.4 Az automatikus modell-generálás algoritmusa 52
6.5 Speciális esetek 56
6.6 Megvalósítás 57
6.7 Alkalmazási eredmények 57

7. NAGY MEGBÍZHATÓSÁGÚ RENDSZEREK KÖZLEKEDÉSI ALKALMAZÁSAI 59
7.1 Bevezetés 59
7.2 A vasúti biztosítóberendezések vizsgálati eljárásai 59
7.2.1 Determinisztikus vizsgálat 60
7.2.2 Determinisztikus vizsgálat valószínűségi adatokkal 61
7.2.3 Valószínűségi alapú vizsgálatok 62
7.3 Valószínűségi határértékek képzése a biztosítóberendezések valószínűségi alapú vizsgálatához 63
7.3.1 Definíciók 63
7.3.2 Az elfogadható kockázati szintből származtatás módszere 64
7.3.3 A már minősített, üzemelő rendszerek megbízhatósági szintjeinek meghatározása 65
7.3.4 A már üzemelő berendezések nem valószínűségi alapú vizsgálati módszereiből származtatással 66

8. ZÁRSZÓ 78

9. RÖVIDÍTÉSEK JEGYZÉKE 79

10. IRODALOMJEGYZÉK 80

11. MELLÉKLETEK 85

1. ELŐSZÓ

1.1 Igény a nagy megbízhatóságra

Napjainkban az élet minden területén az ember életét, munkáját segítő gépek, berendezések alkalmazására kerül sor. A technika fejlődésével ezek a berendezések egyre bonyolultabbak, egyre több elektronikus elemet tartalmaznak.

Az alkalmazások kezdetétől fontos szempont volt egy berendezés megítélésénél az elvégzett funkciók ismerete mellett a minőség megítélése is. A minőség sok paraméter együttese, ezek közül egy – sok esetben az egyik legjelentősebb – a berendezés megbízhatósága. Megbízhatóság alatt azt a valószínűséget értjük, amivel a kérdéses berendezés a definiált funkcióit végrehajtja.

A közlekedés szinte minden területén és az ipar egyes területein kiemelten fontos a nagy megbízhatóság. Ezeken a területeken a berendezések hibás funkció végrehajtásának hatása emberéleteket vagy nagy értékű anyagi javakat veszélyeztet. Ilyen területekre példa a közlekedésnél a vasúti és a légi közlekedés, míg ipari rendszereknél a nukleáris erőművek működése, valamint egyes vegyi üzemekben lejátszódó folyamatok.

A nagy megbízhatóság, mint a minőség egyik aspektusának elérését a tervezés során már figyelembe kell venni, speciális rendszerstruktúrák alkalmazásával kell biztosítani a kívánt megbízhatósági szintet. Ugyanakkor a fentiekben felsorolt alkalmazási területeken – a tervezői hibák, tévedések kiküszöbölése érdekében – nem elégszenek meg a tervezői erőfeszítésekkel, hanem független szakértők ellenőrző munkáját is igénybe veszik. Ilyenkor a független szakértő többek között megbízhatóság-analízist végez a rendszerre, keresvén annak esetleges gyenge pontjait. Ez a független személy által készített elemzés többnyire a hatósági engedélyezés feltétele is. A megbízhatóság-analízis ennek megfelelően fontos szerepet tölt be a rendszertervezésben, és a megbízhatóság kezelése, elemzése külön tudományággá nőtte ki magát. Fontosságát jelzi az évenként megrendezésre kerülő számos nemzetközi konferencia, az igen nagyszámú nemzetközi irodalom.

Ugyanakkor a megbízhatósággal foglalkozó szakembereknek a technika fejlődése következtében mindig újabb és újabb kihívásokkal kell szembenézniük. Az elektronikus eszközök fejlődése részben olyan nagyságú, részben olyan új funkcionalitással rendelkező rendszerek létrehozását teszi lehetővé, amelyek a korábbi analízismódszerekkel nem, vagy nem hatékonyan elemezhetőek. Ugyanakkor az elméleti eredmények gyakorlati alkalmazása szintén nem problémamentes és számos megoldandó kérdést vet fel.

Jelen disszertációban egy, néhány éve bevezetett rendszermegvalósítás, a meghibásodás szempontjából vett adaptív viselkedés analízislehetőségeit vizsgáljuk meg. Elemezzük a már meglévő módszerek alkalmazási lehetőségeit és a rendszerek egy osztályára egyszerűbb analízist lehetővé tévő módszert mutatunk be. Megvizsgáljuk az automatikus analízis lehetőségét, majd elemezzük a módszerek alkalmazhatóságának feltételeit a vasúti rendszereknél.

1.2 Motivációs háttér

A kutatómunka elindítását egyrészről a közlekedési ágazat nagy megbízhatóságú rendszerek iránti igénye motiválta. A közlekedés mindig is az egyik legveszélyesebb üzem volt, a kockázatok csökkentése régen is és napjainkban is folyamatosan napirenden van. A BME Közlekedésautomatikai Tanszék elsősorban a vasúti ágazat biztonsági kérdéseivel foglalkozik. Az ezen a területen alkalmazott biztosítóberendezési technika az ipari vezérléstechnikai terület egyik legösszetettebb, legnagyobb biztonsági igényű része. A biztosítóberendezések alkalmazhatóságának vizsgálata, a sikeres vizsgálat eredményeit összefoglaló alkalmassági tanúsítványok kiadása régóta tanszékünk legjelentősebb ipari megbízásai közé tartoznak. Ezek a munkák igényelték a legújabb európai szabványok alkalmazását is, és ezen keresztül lehetőség nyílott az ezzel kapcsolatos adaptációs vélemények és problémák megismerésére.

A disszertációban publikált kutatásokat másrészről egy valós rendszer, illetve az ott felmerült analízis igénye motiválta. A Paksi Atomerőmű Zrt. a folyamatos biztonsági fejlesztések keretén belül 1995-től folyamatosan terveztette át és cseréltette le az erőművi blokkok biztonsági felügyeletét ellátó védelmi rendszereket. Az új védelmi rendszerek a korszerű processzoros technikán alapulnak, nagy integráltságúak, számos korszerű szolgáltatást képesek nyújtani. Ugyanakkor a korábban leírtaknak megfelelően itt is igény volt, hogy a bevezetésre kerülő rendszert független szakértő cég vizsgálja felül. Ezt a feladatot az MTA-SzTAKI Rendszer- és Irányításelméleti Kutató Laboratóriuma kapta. A felülvizsgálat egyik része volt a rendszer valószínűségi alapú megbízhatóság-analízise is, melyben a szerző, mint külső szakértő vett részt. A védelmi rendszer módosításai a mai napig újabb és újabb elemzési feladatokat is igényelnek.

A disszertációban megjelenő eredmények kötődnek még a BME Elektronikus Jármű- és Járműirányítási Tudásközpont (EJJT) munkájához. A közúti közlekedésben részt vevő járművek is egy biztonságkritikus folyamat részesei, így egyes rendszereik (különös tekintettel a napjainkban egyre nagyobb teret kapó elektronikus fékrendszerekre és elektronikus kormányrendszerek tervezésére) megvalósításánál, üzemeltetésénél a kockázati alapú követelményállítás, a követelményeknek való megfelelés és a megfelelés megbízhatósági alapú bizonyítása kiemelten fontos. Ez a tény motiválta az EJJT vezetését is, amikor az EJJT indulásakor, 2005-ben önálló projektet indított a "járműrendszerek biztonsági szintjének meghatározása" témaelnevezéssel. Ebben a projektben, mint a projekt vezetője vett részt a disszertáció szerzője.

1.3 A disszertáció felépítése

A disszertáció felépítése a következő struktúrát követi: az első fejezet a munka motivációs hátterét bemutatva bevezetőként szolgál. A második fejezet a tényleges szakmai bevezetés. Ebben a fejezetben kerül bemutatásra a disszertációban leírt eredmények kapcsolata a nemzetközi kutatásokkal. Itt röviden bemutatjuk azokat a korábbi eredményeket, amelyekre a munka támaszkodik. A harmadik fejezetben a megbízhatósággal kapcsolatos alapfogalmakat és analízis technikákat foglaltuk össze, különös tekintettel a hibafa-analízisre, amelyet a további vizsgálatokhoz alapmódszernek választottunk. A negyedik fejezet bemutatja a korszerű elektronikus

vezérlőberendezésekben alkalmazható hiba-adaptív viselkedésmódot. Ebben a fejezetben kerül sor a hagyományos analízis módszerek adaptív rendszerekre való alkalmazhatóságának elemzésére, valamint az újonnan kifejlesztett zárt képletek alapján történő számítás módszerének bemutatására. Az ötödik fejezet a megbízhatóság időbeli változásával foglalkozik: tárgyalja a klasszikus időbeli változást leíró modelleket és bemutatja az adaptív viselkedés elemzéséhez is szükséges komplex modelleket. A hatodik fejezet az automatikus hibamodell-létrehozás kérdéskörét öleli fel: vezérlőrendszerek automatikus elemzéséhez ajánl hibafa-generáló algoritmust. A hetedik fejezet foglalkozik a közlekedési ágazat megbízhatósági kérdéseivel, különös tekintettel a vasúti biztosítóberendezések megbízhatóság-analízisére. Végezetül a nyolcadik fejezetben röviden összefoglaljuk a munka eredményeit és szólunk a továbblépési lehetőségekről.

1.4 Köszönetnyilvánítás

Első helyen köszönettel tartozom dr. Bokor József akadémikus úrnak, a Közlekedésautomatikai Tanszék vezetőjének a szakmai és emberi támogatásáért. Már magát a disszertációt is az általa az MTA-SzTAKI kutatócsoportjában felkínált munkalehetőség indította el. Köszönöm a bizalmát, ami lehetővé tette, hogy kezdőként ilyen nagy volumenű, kiemelt figyelmet élvező munka részese lehettem. Ugyancsak sokat adott számomra az a lehetőség, hogy nemzetközi konferenciákon vehettem részt szerte a világban, kicsit belepillantva, belekapcsolódva a megbízhatósági tudománykörrel foglalkozó szakma mindennapi életébe.

Köszönettel tartozom dr. Gáspár Péter úrnak, az MTA-SzTAKI főmunkatársának, akivel pályám korai szakaszában majdnem négy éven keresztül együtt dolgoztunk megbízhatósági és általános rendszerelemzési, rendszertesztelési munkákon a Paksi Atomerőmű Zrt. számára, és akitől a közös munka során sokat tanultam. Pétertől tanultam meg a tudományos cikkírás fortélyait csakúgy, mint egy téma tudományos igényű kidolgozásának módszereit. A szerencsének köszönhetően a későbbiekben is részt vehettem az általa irányított projektekben, és tanácsaival, véleményével, illetve e disszertáció munkahelyi vitához kapcsolódó előbírálatával tudományos tevékenységemet is mindig segítette.

Ez úton mondok köszönetet dr. Gyenes Károly docens úrnak, akivel biztonságkritikus vasúti berendezések fejlesztésén és független szakértői munkáiban dolgoztam együtt, és aki e munkákon keresztül egy általam addig nem művelt területet, a biztonságkritikus szoftverek fejlesztésének területét nyitotta meg számomra. Köszönöm figyelmét, amelyet tudományos munkámnak és e disszertáció megszületésének szentelt, baráti támogatása sok segítséget jelentett csakúgy, mint a munkahelyi vitára elkészített előbírálatának szakmai észrevételei.

Ugyancsak köszönettel tartozom dr. Tarnai Géza egyetemi tanár úrnak, akivel a megbízhatóság-analízis vasúti alkalmazási lehetőségeit vizsgáltuk. Emellett köszönetet mondok azért, hogy részese lehettem a Tata vasútállomásra telepített első magyarországi elektronikus biztosítóberendezés fejlesztési és adaptációs munkáinak, amelyből szakmailag sokat profitáltam, és köszönöm e disszertáció korai verzióihoz fűzött értékes megjegyzéseit.

Végezetül szeretnék köszönetet mondani azoknak a kollégáimnak, akikkel szűkebb szakmai közegben együttdolgozni lehetőségem nyílt. Véleményem szerint a

szakmai fejlődés nem csak az egyén ambícióin és szorgalmán, valamint tudásán múlik: nagyban befolyásolja azt a szakmai közeg, amelyben működik. A BME Közlekedésautomatikai Tanszék munkatársaival, illetve az MTA SzTAKI Rendszer- és Irányításelméleti Laboratórium munkatársaival való közös munka sok szakmai ismerettel gazdagított, és e mellett igazán kellemessé tette a mindennapi munkát.

2. BEVEZETÉS

A nagy megbízhatóságú rendszerek tervezési és ellenőrzési kérdéseivel (illetve általánosan a rendszerek megbízhatóságával és rendelkezésre állásával) külön tudományterület, a megbízhatóság-elmélet foglalkozik. Ez a tudományág felbontható egyrészt hardver és szoftver kérdéseket taglaló részre. A hardver kérdésekkel foglalkozó szakemberek a fizikai rendszert igyekeznek védetté tenni a véletlenszerűen fellépő (esetleg egy időben többszörösen jelentkező) fizikai meghibásodásoktól. Itt nem csak a megvalósított rendszerstruktúra lehet kritikus, de az alkalmazott hibafeltárási mechanizmusok, a felfedett hibák megszüntetésére tett intézkedések is [Apostolakis et. al.,1978], [Storey, 1996]. A szoftver kérdések között elsősorban a tervezés fázisában bekövetkező emberi hibák és tévesztések hatásainak kivédése a cél. Itt részben a programtervezés- és programfejlesztés fázisában alkalmazott módszerek és eljárások, valamint a felépítendő szoftverstruktúra meghatározása a cél [Bittanti, 1987]. Mindkét terület foglalkozik a tervezési elvek mellett az ellenőrzés módszereivel is, és külön-külön modellezi a fellépő hibákat, rendellenességeket [Tobias és Trindade, 1998]. Mivel a két terület erősen összefügg (pl. szisztematikus hibák jelenléte), a két diszciplína ötvözésével néhány éve jött létre a “hardware-software co-design” néven emlegetett új tudományterület [Csertán et. al., 1996].

A megbízhatóság-elmélet egy másik megközelítés szerint felbontható tervezési és megbízhatóság-analízis részekre. Ennek a felbontásnak az előnye kettős: egyrészt együtt kezeli a hardver és a szoftver objektumokat, másrészt igazodik a tervezés során alkalmazott feladatmegosztáshoz is. A fenti felbontás tervezési részénél nehéz általános érvényű szabályokat adni a rendszerek létrehozásához, éppen ezért szakirodalom szintjén is igen kevés forrás említhető. Ezzel szemben az egyedileg tervezett rendszer jól analizálható általános módszerekkel, így a megbízhatóság-analízis tudományterület igen aktív.

A megbízhatóság-analízis technikák közül napjainkban az egyik legelterjedtebben alkalmazott módszer a hibafa-analízis. Az eljárás kiforrott módszertannal és hatékony szoftver eszközökkel rendelkezik [Chunning és Dinghua, 1990]. A hibafa-analízis kutatások napjainkban újabb és újabb, eddig nem vagy csak körülményesen kezelhető területeket vonnak be az analízisbe. A hagyományos kétállapotú modellel dolgozó hibafa-analízis kiterjesztésre került többállapotú rendszerekhez [Aven, 1985]. Napjaink kutatásaiban a hangsúly a dinamikus rendszerviselkedések modellezésére helyeződött. Ilyen dinamikus viselkedési módot képviselnek a hidegtartalékok esetei, a késleltetetten bekövetkező események vagy a számítógépek pillanatnyi hiba miatti újraindulásai is [Dugan et. al., 1990]. A disszertációban tárgyalt hiba-adaptív rendszerkomponensek szintén a dinamikus viselkedésű objektumokhoz tartoznak, és talán mivel alkalmazásuk még nem terjedt el széleskörűen, analízisük sem megoldott.

Ugyanakkor a gazdasági szempontok fokozottan igénylik az erőforrásokkal való takarékosságot, így a karbantartás optimalizálása is előtérbe került. Ennek következményeként az analíziseknek nem csak a rendszer megbízhatóságának átlagos értékeit, hanem időbeli változását is szolgáltatniuk kell. A korábban említett újabb területek analízisbe való bevonása igényli a megfelelő időfüggő modellek kidolgozását is.

Szintén részben gazdasági szempontú az analízis automatizálásának igénye. A megbízhatóság-analízis modell létrehozási és elemzési fázisokból áll. A nehézség a modell létrehozási fázisban jelentkezik, itt a tervezéshez való szoros kötődés miatt nehéz általános módszereket létrehozni [Kocza és Bossche, 1997].

Jelen disszertáció a megbízhatóság-analízis témakörén belül három részproblémára összpontosít: 1. Definiálja a hiba-adaptív funkciókat, bemutatja megbízhatósági szempontból vett elemzésük lehetőségeit, és analízis-kiterjesztést javasol egységes kezelhetőségük érdekében [Szabó és Gáspár, 1999b]; 2. A hiba-adaptív funkciókhoz kapcsolódóan komplex megbízhatósági időfüggési modelleket vezet be [Gáspár és Szabó, 1998b]; 3. Szoftver alapú rendszerek hibafa-analíziséhez automatikus modellgenerálási algoritmust javasol [Gáspár és Szabó, 1999b].

A megbízhatóság-elmélet alkalmazási területei elsősorban a nukleáris erőművek és a közlekedés különböző ágazatai. A közlekedés ágazataiban alkalmazott rendszerek közül a vasúti biztosítóberendezések azok, amelyek kiemelten igénylik a nagy megbízhatóságra való tervezés alkalmazását, és amelyeknél a megvalósított biztonsági szint bizonyítása is elengedhetetlen [Hudoklin és Rozman, 1985]. Minden megbízhatósági vizsgálat kritikus pontja az elfogadási kritérium meghatározása. Vasúti alkalmazások esetén az elfogadási szint származtatására többféle lehetőség kínálkozik, ezeket a lehetőségeket elemzi a disszertáció [Szabó és Tarnai, 1999].

3. A MEGBÍZHATÓSÁGI SZINT DEFINIÁLÁSA ÉS VIZSGÁLATA

3.1 Definíciók

Az alábbiakban felsoroljuk a megbízhatóság-elmélet legfontosabb alapfogalmait. Az egyébként sokszor sziporkázóan gazdag magyar nyelvben sajnos még nem alakult ki minden egyes, megbízhatósághoz kapcsolódó fogalom magyar megfelelője (ennek oka részben a tudományterület relatíve fiatal volta, részben a Magyarországon való szűk körű alkalmazásban keresendő), ezért ahol csak lehetséges, utalunk a fogalomkör angol megfelelőjére [Apostolakis et. al.,1978], [Schaefer, 1983], [Storey, 1996], [Tarnai d].

Az általános értelemben vett megbízhatóság (dependability) magában foglalja egy rendszer összes olyan tulajdonságát, amely a nagy megbízhatósági szint elérésére vagy fenntartására hatással lehet. A magyar terminológia: “nagy megbízhatóságú rendszer” is az angol “high dependable system” megnevezésnek felel meg. Az általános értelemben vett megbízhatóság része a minőségnek (quality), de nem minden összetevőjét foglalja magában.

Az általános értelemben vett megbízhatóság fontosabb összetevői a következők:

- Megbízhatóság (szűk értelmű), vagy más néven működőképesség (reliability),
- Rendelkezésre állás (availability),
- Biztonság (safety),
- Karbantarthatóság, javíthatóság (maintenability),
- Tesztelhetőség (testability),
- Áttekinthetőség, megismerhetőség (recognizability),
- Kezelhetőség (operability),
- Védettség (security).

Az általános értelemben vett biztonsági jellemzőkre szokás a RAMS betűszóval is utalni, amely a reliability, availability, maintaenability és a safety angol szavak kezdőbetüiből képződik.

Megbízhatóság (működőképesség): Egy berendezés azon tulajdonsága, amely azt jellemzi, hogy a berendezés mennyiben képes a számára előírt követelmények megadott határokon belüli, meghatározott időtartamon keresztüli teljesítésére. Valószínűségként szokás definiálni.

A fenti jellemző más megfogalmazása szerint annak a valószínűségét takarja, hogy a vizsgált berendezés az üzembe helyezésétől a vizsgálat időpontjáig nem hibásodik meg.

Rendelkezésre állás: Annak a valószínűsége, hogy a vizsgált berendezés a *t* időpillanatban működőképes. Amennyiben a rendszerben javítást nem alkalmazunk, ez megegyezik a működőképességgel.

Biztonság: Egy berendezés azon tulajdonsága, amely azt jellemzi, hogy a berendezés mennyiben képes a működése során az emberi életre vagy az anyagi javakra veszélyes állapotok elkerülésére. A biztonságot sokszor a veszélyeztetettség hiányaként definiálják, ahol a veszélyeztetettség olyan szituáció, amelyben lehetőség van emberi életre vagy anyagi javakra veszélyes szituáció bekövetkezésére. Valószínűségként szokás definiálni.

Karbantarthatóság: A rendszer azon tulajdonsága, hogy leállás (meghibásodás) esetén mennyiben és milyen időtartam alatt hozható ismét működőképes állapotba. A karbantarthatóság sokszor determinisztikusan vizsgált fogalom, ilyenkor a rendszert karbantarthatónak (javíthatónak) mondjuk, ha tetszőleges meghibásodása után a rendelkezésre álló erőforrásokkal (személyzet, eszközök, tartalék alkatrészek) egy előre definiált idő letelte előtt újra működőképes állapotba hozható. Definiálható a karbantarthatóság valószínűségként is, ekkor annak a valószínűséget jelenti, hogy a rendszer egy tetszőleges meghibásodása után a rendelkezésre álló erőforrásokkal egy előre definiált idő alatt újra üzemképes állapotba hozható. A karbantarthatóság fogalma a rendszer tervezési paraméterei mellett a rendelkezésre álló személyzet képzettségének a színvonalát, a rendelkezésre álló pótalkatrész elégséges voltát stb. is tartalmazza.

Tesztelhetőség: A rendszer azon tulajdonsága, hogy a lehetséges meghibásodásait akár öntesztekkel, akár külső tesztek (manuális beavatkozás) útján képes felfedni. A tesztelhetőséget kétféleképpen szokás megadni: megadható az összes lehetséges meghibásodás közül a felfedhetők százalékos arányának kifejezésével, illetve megadható adott időegység alatt fellépő meghibásodások közül felfedhetők százalékos arányának kifejezésével.

Áttekinthetőség: A rendszer azon tulajdonsága, amely azt jellemzi, hogy működését, karbantartását, javítási metódusait mennyire könnyű megérteni. A rendszer tervezési paraméterei mellett a dokumentáció milyenségét is tartalmazza.

Kezelhetőség: A rendszer azon tulajdonsága, amely azt jellemzi, hogy működtetése milyen beavatkozásokat, kezeléseket igényel, a szükséges kezelések végrehajtása egyértelmű-e, ill. felcserélhető-e stb.

Védettség: A rendszer azon tulajdonsága, amely azt jellemzi, hogy működtetése során mennyire képes megakadályozni a jogosulatlan hozzáféréseket, szándékolatlan vagy rosszindulatú beavatkozásokat.

A nagy megbízhatóságú rendszerek létrehozásának célja minden esetben a külső vagy belső veszélyforrások hatásainak elhárítása. A berendezés működését befolyásoló külső veszélyforrásokkal (pl. földrengés) a továbbiakban nem foglalkozunk, csak a belső veszélyforrásokat elemezzük.

A berendezésben (számunkra károsan) hibák (fault) keletkeznek. A hiba a berendezés valamely paraméterének nem megengedett eltérése a névleges értéktől. Hatása, fennállása lehet időszakos, ekkor zavarról beszélünk, vagy lehet állandó, ekkor változásnak nevezzük. Amennyiben a paraméter eltérés a funkció-végrehajtásban is jelentkezik, meghibásodás (failure) következett be. A meghibásodás definíciója: a berendezés működésében olyan esemény következett be, amely a funkcionalitásának teljes vagy részleges elvesztését vonta maga után.

A berendezésben keletkező meghibásodások okai eredhetnek emberi hibákból (human error), amelyeket a rendszer tervezése, kivitelezése, kezelése és fenntartása során lehet elkövetni. Az emberi hibák, illetve viselkedés modellezése külön terület, ezzel a továbbiakban részletesen nem kívánunk foglalkozni.

A rendszerek megbízhatóságának jellemzésére alkalmazott paraméterek nagy részénél utaltunk arra, hogy a paraméter többnyire valószínűségként van definiálva. A rendszerek megbízhatóságának jellemzésére nem csak a valószínűségi paraméterek kínálkoznak. A mérnöki gyakorlat számára sokszor jobban megfogható jellemzést adnak az átlagos értékek.

Meghibásodási gyakoriság (failure rate): Megadja, hogy egy adott időegység alatt a vizsgált rendszer átlagosan hányszor hibásodik meg. Mértékegysége 1/óra, de gyakran használt egység a FIT is (FIT – Failures In Time – 10^9 óra alatt átlagosan bekövetkező meghibásodások száma).

Meghibásodások közötti átlagos idő (Mean Time Between Failures, MTBF): Megadja, hogy a rendszer két meghibásodása között átlagosan mekkora idő telik el.

Átlagos javítási idő (Mean Time To Repair, MTTR): megadja, hogy a rendszer meghibásodása utáni újbóli üzembeállítás átlagosan mekkora időt vesz igénybe.

Meg kell jegyezni, hogy a gyakorlatban alkalmazott rendszerek nagy része javítható rendszer, vagyis egy meghibásodás után javítás segítségével a rendszer tovább üzemeltethető. Vannak azonban nem javítható (küldetéskritikus - mission critical) rendszerek is, amelyekben a teljes rendszer meghibásodása után már nem állítható vissza a funkcionalitás. Ilyen rendszerek pl. a személyzet nélküli rakéták és űrrakéták vezérlő elektronikái, de sok esetben ilyenek a repülőgépek elektronikai berendezései (és mechanikai berendezései) is, hiszen még ha elméletileg lehetséges is a javítás, de hiányoznak a személyi és a tárgyi feltételei. Ezekben a rendszerekben az MTBF érték helyett a meghibásodásig tartó átlagos idő (Mean Time To Failure – MTTF) fogalmát, valamint a rendelkezésre állás helyett a megbízhatóság (itt túlélési valószínűségnek is nevezett) fogalmát alkalmazzák.

3.2 A biztonsági (RAMS) paraméterek specifikálása és kezelése

Természetes igény, hogy a rendszerek biztonsági paramétereivel szembeni elvárásainkat specifikáljuk, illetve ezen biztonsági követelményeknek a teljesülését vizsgáljuk.

A specifikáció kérdéskörével sok hazai és külföldi publikáció foglalkozik pl. [Szabó és Tarnai, 2002], [Szabó és Tarnai, 2003], csakúgy, mint a specifikált értékek teljesítésének lehetőségével pl. [Szabó, 2007a], [Szabó, 2007b], [Szabó et. al., 2004]. E területekre általános, szakterület-független szabványok csakúgy léteznek [IEC 61508], mint szakterületfüggő előírások, pl. a vasúti közlekedés számára [EN 50126], [EN 50129].

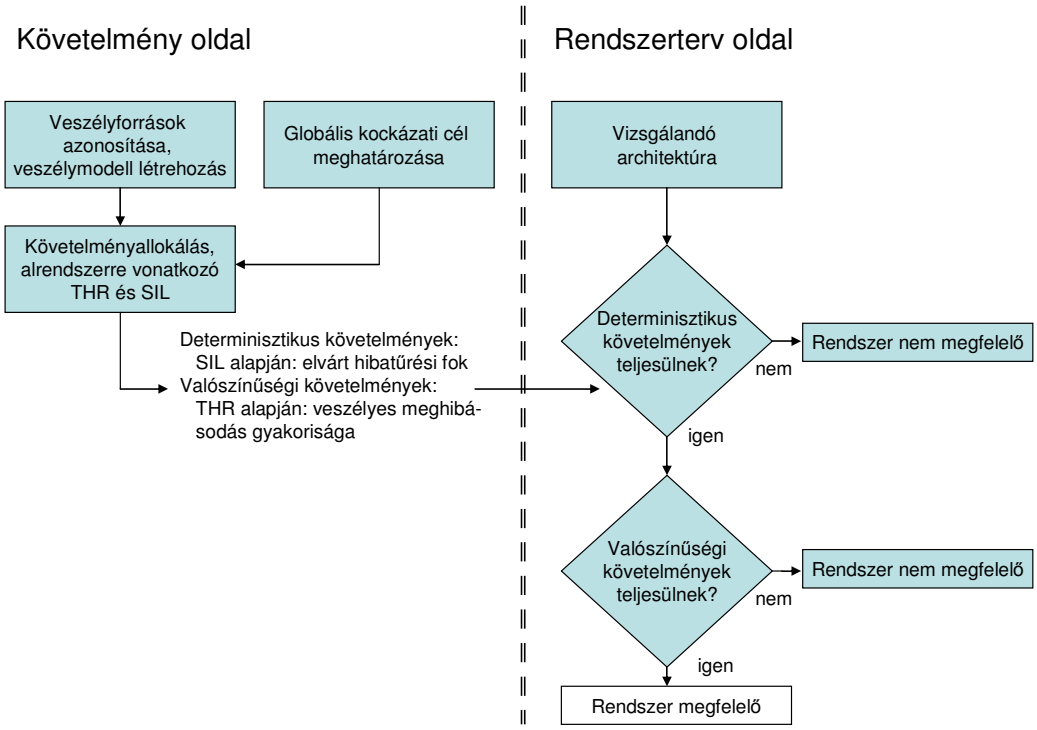
Az alábbiakban röviden összefoglaljuk a biztonsági követelmények kezelésének lehetséges folyamatát [Szabó, 2008].

Bármely rendszer tervezésének alapja a többnyire a megrendelő által szolgáltatott specifikáció. (Szokás a specifikációt, mint a megrendelő és a szállító közös megegyezésén alapuló alapdokumentumot tekinteni.) A specifikáció általános rendszerek esetén két részből áll: a funkcionális specifikáció deklarálja a rendszer működésével, viselkedésével szemben támasztott követelményeket, míg a műszaki specifikáció írja le a környezettel való kapcsolat követelményeit. Nagy megbízhatóságú rendszerek esetén a fenti két specifikációt kiegészíti egy harmadik, a biztonsági specifikáció, amely a berendezés elvárt megbízhatósági paramétereit specifikálja.

Mindenképpen szükséges kettéválasztanunk a kezelési folyamatot egy specifikációs és egy specifikációt teljesítő, illetve igazoló szakaszra (3.1 ábra). A biztonsági specifikáció praktikusan a tervezendő rendszer funkcióiból indul ki. A funkciók elvárt viselkedésétől való eltérés következményeinek és gyakoriságának elemzésével (kockázatelemzés), valamint egy globális kockázati céllal való összevetésével (kockázatértékelés) meghatározható a rendszer által okozott veszélyeztetések eltűrhető gyakorisága (THR - Tolerable Hazard Rate).

A THR követelmény valószínűségi alapú követelmény és alapvetően a rendszer architektúra megfelelő megválasztásával, az alkalmazott alkatelemek megbízhatósági szempontú megválasztásával, valamint a szükséges meghibásodást felfedő tesztek ciklusidejének megválasztásával teljesíthető.

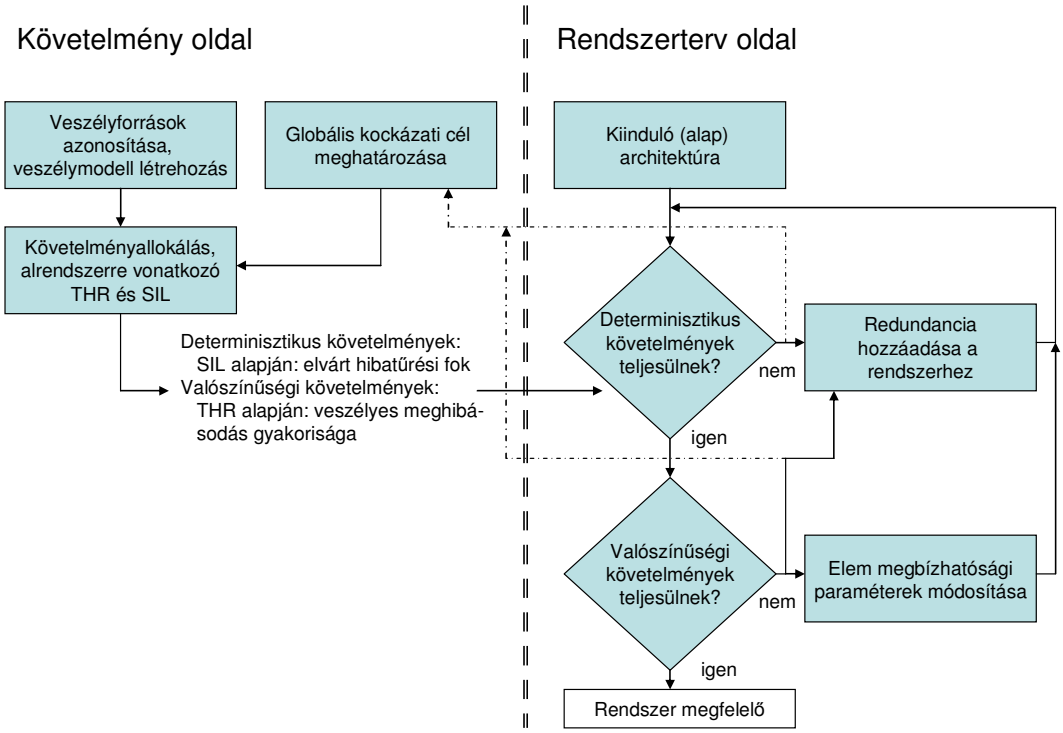
Ugyanakkor egy rendszerrel szemben szokás determinisztikus biztonsági követelményeket állítani, részben tapasztalati, részben a valószínűségi alapú kritériumokon alapulva, azokhoz kötve (pl. egyszeres hibatűrés vagy elvárt javítási idő). A determinisztikus követelmények kategóriájába tartozik a teljes életciklus alatt a rendszerbe bekerülő, emberi okú hibák (pl. specifikációs hiba, tervezési hiba, üzemeltetési hiba stb.) elleni elvárt védettség szintje is. Ez a szint a SIL (Safety Integrity Level) és értékét a meghatározott THR követelmény alapján határozzák meg.



3-1. ábra: A biztonsági követelmények specifikálása és a teljesülés vizsgálata

A specifikált biztonsági követelmények teljesítése is kettős: teljesíteni kell a determinisztikus (hibatűrésre vonatkozó) követelményeket - ezt alapvetően a rendszerben alkalmazott redundancia fokával, illetve a fokszám növelésével lehet megtenni; illetve teljesíteni kell a determinisztikus (veszélyes meghibásodások gyakoriságára vonatkozó) követelményeket - ezt viszont több, konkurens úton is teljesíteni lehet: akár a redundancia növelésével, akár az egyes elemek megbízhatósági paramétereinek módosításával (megbízhatóbb elemek választásával, igénybevétel csökkentésével, a tesztelési/hibafelfedési gyakoriság növelésével). És meg kell említeni egy harmadik lehetőséget is: a követelmények módosítását. A kockázati követelmények végső soron gazdasági követelmények: azt kell eldönteni, mennyit ér egy adott biztonsági szint. Előfordulhat, hogy a rendszer létrehozása során szembesülünk ezzel (lásd 3-2. ábra).

A disszertáció a fenti specifikációs, teljesítő és igazoló folyamat bizonyos elemeihez kíván hozzájárulni. Az első tézisben megfogalmazott, hiba-adaptív és hiba-aktív logikák elemzésére vonatkozó módszer a rendszerek megbízhatósági elemzésénél (mind determinisztikus, mind valószínűségi alapú elemzések) alkalmazható. A második, a rendszerek megbízhatósági jellemzőinek időfüggésére vonatkozó tézis a valószínűségi követelmények teljesülésének vizsgálatára, illetve az optimális teszt ciklusidő meghatározásához használható. A harmadik tézisben megfogalmazott automatikus hibafa-generálási algoritmus az iteratív tervezések során sokszor, kis módosításokkal ismétlődő elemzéseket segíti az automatikus hibamodell generálással, illetve ezekből a korábban emberi erőforrással végrehajtott modellépítési lépésekből szűri ki az emberi hibákat. A negyedik tézisben megfogalmazott állítások a globális kockázati cél kitűzését segítik.



3-2. ábra: A biztonsági követelmények specifikálása és a specifikáció teljesítése

3.3 Analízis eljárások

Az előző részben bemutatott megbízhatósági jellemzők vizsgálata ennek megfelelően igen fontos feladat már a tervezés fázisában is, hiszen a gyártónak biztosítania, majd az átadáskor bizonyítania kell a biztonsági specifikáció teljesülését is. Azonban amíg a funkcionális és a műszaki specifikáció teljesülése tesztekkel könnyen igazolható, a megbízhatósági paraméterek teszteléses vizsgálata igen hosszú időt és nagy anyagi ráfordítást igényel, mivel a meghibásodásokat is tartalmazó rendszerkonfigurációkat kell a tesztelés számára előállítani. Éppen ezért a biztonsági specifikáció teljesülését többnyire elméleti analízisekkel végzik el, amelyeknek csak az alapadatai származnak specifikus tesztekből vagy a valós életből.

Az analízis-eljárásokat két csoportra lehet bontani: az egyiket az angol terminológia kvalitatív (minőségi) míg a másikat kvantitatív (mennyiségi) módszereknek nevezi. Az angol terminológiának megfelelően pl. vasúti területen lehet találkozni a mennyiségi biztonságigazolás és a minőségi biztonságigazolás fogalmával. Jelen munkában azonban inkább a módszereket jobban jellemző (noha nem a szó szerinti fordításból származó) determinisztikus és valószínűségi alapú vizsgálatokról fogunk beszélni.

A két módszer-csoport az alábbiak szerint jellemezhető: determinisztikus vizsgálat esetén egy megválaszolandó (esetleg az analízist végző személytől függően skálázandó) kérdésre keressük a választ. Ilyen például a rendszerekkel szemben gyakran támasztott egyszeres meghibásodás kritériuma. Ez a feltétel azt írja elő a rendszerrel szemben, hogy bármely komponensének meghibásodásával szemben a rendszer hibatűrő legyen, amennyiben csak az az egy meghibásodás aktív az adott pillanatban. Ennek a kritériumnak a vizsgálata „teljesíti” – „nem teljesíti” eredménnyel zárulhat csak. Ezzel szemben valószínűségi alapú vizsgálatnál a rendszert egy (vagy

több) valószínűségi paraméterrel jellemezzük (MTBF, rendelkezésre állás stb.). A valószínűségi vizsgálatoknál a rendszerkomponensekről valószínűségi alapadatokkal kell rendelkeznünk.

A fejezet további részében röviden bemutatjuk a legfontosabb megbízhatóság-analízis eljárásokat [Leveson, 1995], [Apostolakis et. al.,1978], [Hwang et. al., 1981], [Gáspár és Szabó, 1998c]. A következő módszerek kerülnek bemutatásra:

- Becslési eljárások,
- Ellenőrzési listák és kockázati indexek,
- Meghibásodási módok és hatások analízise,
- Eseményfa analízis,
- Hibafa analízis,
- Markov analízis.

3.4 Becslési eljárások

3.4.1 Bevezetés

A becslési eljárások célja egy eszköz vagy egység meghibásodási rátájának (esetleg további jellemzőinek, mint pl. az első év szorzófaktora, a meghibásodási ráta időfüggvénye) meghatározása. A módszer története viszonylag rövid, 1975 környékén került először alkalmazásra [Jain,1997], [RPP Bellcore], [MIL-HDBK 217F].

A becslési eljárások minden hardver rendszert soros rendszerként kezelnek, nem veszik figyelembe (illetve nem képesek figyelembe venni) a rendszerben megvalósított redundanciát. A meghibásodás szempontjából soros rendszerben bármely elem meghibásodása a teljes rendszer meghibásodását vonja maga után, és a rendszer meghibásodási rátája az elemek meghibásodási rátáinak összegeként számítható. Becslés alkalmazható szoftver rendszerekre is, és létezik módszertana mechanikai elemekre is [SRP Bellcore].

Becsléssel elsősorban olyan eszközök meghibásodási rátáját szokás számítani, amelyekett egy magasabb szintű analízisben (pl. hibafa-analízis) már nem kívánunk részletesen elemezni, hanem un. alapeseményként tekintünk (a becslési módszerek alkalmazásához lásd a 4. mellékletben található példát).

3.4.2 Alkatrész számlálás módszere

Az eljárás alapmódszertana az úgynevezett alkatrész-számláláson alapszik: Soros rendszerként kezelve a vizsgált objektumot, az objektum meghibásodási rátája az egyes építőelemek (alkatrészek) meghibásodási rátáinak összegeként kerül számításra. Az építőelemek meghibásodási rátáit tipikus elemekre lebontva katalógusok tartalmazzák, amelyek a módszert leíró ajánlások részei.

A valóságot messzemenően közelítő eredmény eléréséhez szükséges figyelembe venni az adott eszköz környezeti igénybevételét és minőségét is. Ezek alapján az alkatrész számlálási módszer az alábbi képlettel írható le:

$$\lambda_e = \Pi_e \cdot \sum_i \lambda_{gi} \cdot \Pi_{Ti} \cdot \Pi_{Si} \cdot \Pi_{Qi} \qquad \textbf{(3-1.)}$$

ahol λ_e a vizsgált objektum számított meghibásodási rátája, Π_e az objektum környezeti faktora, λ_{gi} az i. elem általános meghibásodási rátája (katalógusadat), Π_{Ti} az i. elem hőmérsékleti faktora, Π_{Si} az i. elem elektromos igénybevételi faktora, és Π_{Qi} az i. elem minőségi faktora.

A környezeti faktor értéke 1, ha a vizsgált objektum nem mobil, földi felhasználású, és növekszik mobil felhasználás esetén.

A hőmérsékleti faktor az Arrhenius modell alapján kerül meghatározásra, amelyet eredetileg vegyi folyamatok jellemzéséhez vezettek be, de az a megfigyelés, hogy a meghibásodások kezelésénél is jól alkalmazható. A modell szerint:

$$\Pi_{Ti} = e^{-\frac{E_a}{k \cdot T^*}} \; ;$$
$$k = 8.62 \cdot 10^{-5} \frac{eV}{K} \quad és \qquad \textbf{(3-2.)}$$
$$\frac{1}{T^*} = \frac{1}{(T+273)} - \frac{1}{(40+273)}$$

A modellben E_a az aktivizációs energia. A modellből láthatóan T=40°C hőmérsékleten a hőmérsékleti faktor 1, és a hőmérséklet növelésével növekszik (magasabb működési hőmérséklet magasabb igénybevételt jelent).

Az elektromos igénybevételi faktor meghatározásánál azt feltételezik, hogy egy alkatrész határadatokon való üzemeltetése az alkatrész idő előtti meghibásodását eredményezi, valamint azt, hogy 50% az az igénybevétel, amelynél az alkatrész átlagos megbízhatósági adatokat produkál. Csökkentve az igénybevételt, csökken az alkatrész meghibásodási rátája.

$$\Pi_{Si} = e^{m(p_1 - p_0)} \qquad \textbf{(3-3.)}$$

ahol p_0 a referencia-igénybevétel (50%), p_1 a tényleges igénybevétel, m pedig az elektromos faktor, melynek értéke tipikusan 0,025 körüli. A modellből láthatóan 50%-os igénybevételnél az elektromos igénybevételi faktor értéke 1, és az igénybevétel növelésével növekszik.

A minőségi faktor meghatározását az előző két faktorral ellentétben jóval nagyobb szabadságfokkal bízzák az analízist végzőre. Ennek a faktornak a használatát az a tapasztalat indokolja, hogy a széleskörű minőségellenőrzési programmal, nagy gyártási tapasztalattal rendelkező cégek termékei jobb megbízhatósági paramétereket mutatnak, mint az azonos funkciókkal rendelkező, de máshonnan származó termékek. A minőségi faktor alkalmazása, illetve alkalmazási módja mérnöki döntés kérdése.

3.4.3 Laboratóriumi információk figyelembevétele

A becslés alapmódszeréhez képes a valóságot jobban közelítő eredményeket lehet akkor kapni, ha a hipotetikus információk mellett laboratóriumi teszt eredményekkel is rendelkezünk a vizsgált objektum valamely alkatrészéről. Ilyen esetben a laboreredmények súlyozottan vehetők figyelembe, annak függvényében, hogy mekkora mintán történt a vizsgálat, mekkora időtartam alatt történt a vizsgálat stb.

$$\lambda = w \cdot \lambda_g + (1-w) \cdot \lambda_{tab} \qquad \textbf{(3-4.)}$$

A 3-4. képletben λ_g az alkatrész katalógusban megadott meghibásodási rátája, λ_{tab} a laboratóriumi teszt által szolgáltatott meghibásodási ráta, *w* pedig a súlyozó tényező. A módszer olyan esetekben előnyös, amikor nem áll rendelkezésre elégséges vagy megbízható adat az első módszer alkalmazásához, pl. bevezetés alatt álló termékeknél.

3.4.4 Tapasztalati adatok figyelembevétele

Szintén a valóságot jobban közelítő eredmények elérése motiválta a harmadik módszert, amelyben a becsült adatokat a felhasználás során szerzett tapasztalati adatokkal módosítják. A módszer alkalmazható a vizsgált objektum alkatrészeire vagy az alkatrészek egy csoportjára, de gyakoribb alkalmazása, amikor magára az objektumra alkalmazzák.

$$\lambda = w \cdot \lambda_g + (1-w) \cdot \lambda_{valós} \qquad \textbf{(3-5.)}$$

A 3-5. képletben λ_g az alkatrész katalógusban megadott meghibásodási rátája, $\lambda_{valós}$ a korábbi működés alapján kapott meghibásodási ráta, *w* pedig a súlyozó tényező. Nagy megbízhatóságú rendszerek esetén a gyártók minőségbiztosítási programjának szinte minden esetben része a termékek életciklusának követése, így ezek a gyártók nagyszámú üzemórán alapuló megbízhatósági adatokkal rendelkeznek termékeikről. Ezek az adatok alkalmasak a becslés alapmódszerével szerzett információ pontosítására.

3.5 Ellenőrzési listák és kockázati indexek

Az ellenőrzési listák a már megszerzett tapasztalat (tervezői és üzemeltetői) alkalmazásának szisztematikus ellenőrzésére szolgálnak. Folyamatos fejlesztésükkel pontokba szedhető az összes olyan emberi hiba, amely által okozott problémákat célszerű volna elkerülni. A lista többnyire “Mi történik, ha…” típusú kérdéseket, valamint szabványokat és ajánlásokat tartalmaz.

A hardver vonatkozásában az ellenőrzési listák a tesztelések, rendszeres ellenőrzések alkalmával játszanak kiemelkedő szerepet, elősegítve, hogy az előírt, elvégzendő műveletek közül semmi se maradjon elvégzetlenül.

Az ellenőrzési listák használata ugyanakkor komplex rendszerek esetén nehézkes, a kérdések megválaszolására sokszor más analízismódszert kell alkalmazni. Ennek következtében nagy rendszerek esetén az ellenőrzési listák vizsgálati, ellenőrzési forgatókönyvként funkcionálhatnak.

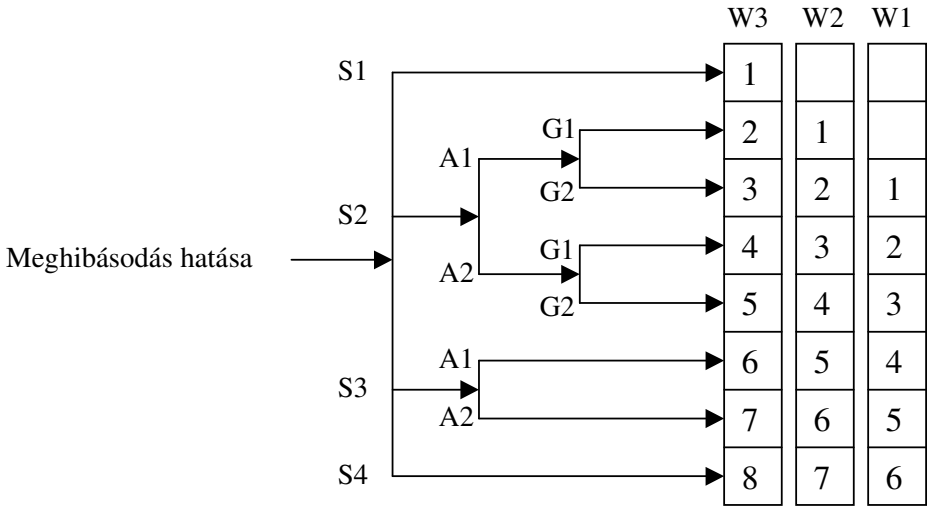
A kockázati indexek komplex rendszerekben fellépő veszélyhelyzetek azonosítására szolgálnak. Elsődlegesen vegyipari létesítmények számára dolgozták ki őket, definiálva az ott fellelhető tipikus veszélyhelyzeteket, de ma már léteznek kockázati indexek különböző ipari területekre.

A kockázati indexek alkalmazásakor a rendszert egységekre bontják, és az egyes egységekben fellelhető, a kockázati indexben felsorolt veszélyhelyzetek alapján rangsorolják az egységeket.

Kockázati indexként ugyancsak alkalmazható a meghibásodások hatásainak kritikusság vizsgálatára (noha elsősorban adott feladat biztonsági követelményeinek osztályba sorolására szolgál) a DIN V 19250 [DIN19250] ipari szabvány által definiált követelmény-osztály rendszer. Az események besorolása itt négy paraméter alkalmazásával történik:

1. A meghibásodás által okozható kár mértéke (4 fokozat, S1-S4),
2. A veszély által érintett zónában való tartózkodás (2 fokozat, A1-A2),
3. A veszély elhárításának lehetősége (2 fokozat, G1-G2), valamint
4. A meghibásodás bekövetkezésének valószínűsége (3 fokozat, W1-W3).

A négy szempont alapján 8 kockázati osztályba sorolhatjuk a meghibásodás hatásait (3-3. ábra), ahol a 8-as a legsúlyosabb kockázati osztály. Ezt a módszert a vasúti biztosítóberendezési technikában is alkalmazzák olyan esetekben, amikor a felhasznált vezérlőberendezés a DIN V 19250 szerinti minősítéssel rendelkezik.



3-3. ábra: DIN V 19250 szerinti kockázati osztályok megállapítása

3.6 Meghibásodási módok és hatások analízise

3.6.1 FMEA jellemzői

A meghibásodási módok és hatások analízise (Failure Mode and Effects Analysis – FMEA) egy tetszőleges rendszer, alrendszer vagy funkció strukturált, minőségi (determinisztikus) analízise, amelynek célja a lehetséges rendszermeghibásodások

felfedése, következményeik és a rendszerműködésre gyakorolt hatásuk feltárása. Gyakran az FMEA módszert kiegészítik a meghibásodási hatás súlyosságának és valószínűségének a meghatározásával – ilyenkor a módszert meghibásodási módok, hatások és kritikusság analízisnek (Failure Mode, Effects and Criticality Analysis – FMECA).

Noha az FMEA eljárást a megbízhatóság-analízis módszerek közé soroljuk, legnagyobb előnye talán abban rejlik, hogy rákényszeríti a vizsgálatot végző személyt a rendszer mély megismerésére. Éppen ezért a tervezés fázisában, iteratív módon is ajánlott használni.

3.6.2 FMEA módszertan

Egy FMEA vizsgálat az alábbi lépésekből tevődik össze:

- Alapszabályok és feltételezések definiálása (rendszer működési fázisok, működési környezet, a működés célja stb.),
- Az analízisvégzés szintjének definiálása (vajon az egész rendszerre, vagy csak egy részére végezzük a vizsgálatot),
- Az egyes analizálandó egységek, alegységek definiálása (alrendszer, modul, funkció, komponens),
- Az összes, vizsgálatban érintett komponens lehetséges meghibásodási módjainak összegyűjtése. Az így nyert listát sok helyen hibakatalógusnak nevezik – ez a hibakatalógus a későbbi vizsgálatokhoz is felhasználható. Egyes területeken az FMEA (vagy ahhoz módszertanban hasonló) eljárások számára szokványos elemek figyelembe veendő meghibásodási módjait ajánlások vagy szabványok definiálják.
- Minden egyes komponens lehetséges meghibásodási módjai következményeinek feltárása,
- A következmények osztályozása a rendszerműködésre gyakorolt hatásuk alapján,
- Az egyes meghibásodási módok detektálhatóságának vizsgálata,
- Amennyiben szükséges, kompenzációs módok vagy tervváltoztatások vizsgálata a veszélyesnek ítélt meghibásodások hatásainak elkerülésére.

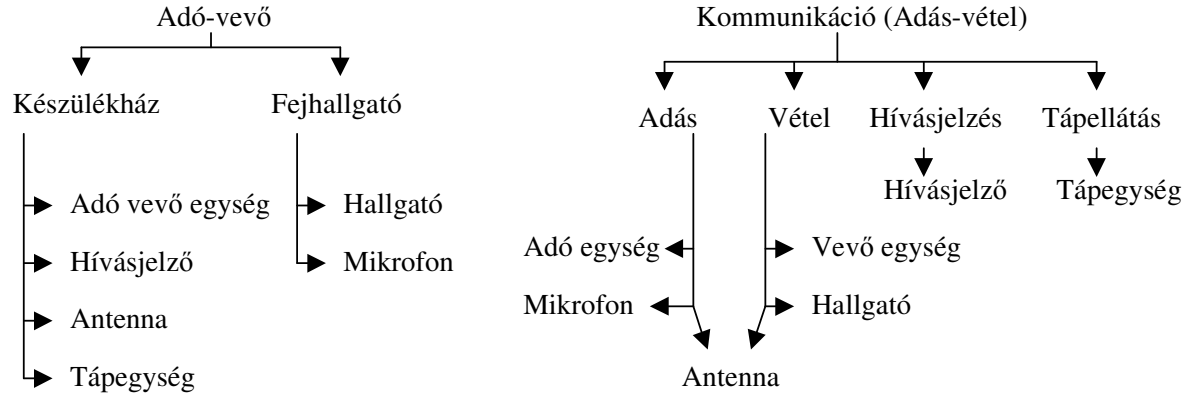
Nagy méretű rendszerek esetén célszerű a vizsgálatot hierarchikusan elvégezni: a rendszer (akár többszintű) alrendszerekre bontása után a hierarchiában alul lévő elemek vagy alegységek elemzése az első lépés a módszertannak megfelelően. Az egyes elem meghibásodási módok alrendszerre gyakorolt hatása alapján megállapíthatóak az alrendszer meghibásodási módjai, amelyeket a hierarchia magasabb szintjén ugyanúgy kezelünk, mint az alacsony szinten a komponens meghibásodási módokat.

Kimondhatjuk tehát azt, hogy az FMEA eljárás bottom-up típusú (alulról indul a vizsgálat felfelé), de nagy rendszereknél ehhez először a rendszer top-down típusú (felülről lefelé történő) strukturálása szükséges. A strukturálás három modell típus szerint történhet:

1. Strukturális modell: Az alegységek, komponensek meghatározásánál azok fizikai megjelenését, szeparáltságát vesszük figyelembe,

2. Funkcionális modell: Az alegységek, komponensek meghatározásánál azok funkcióit vesszük figyelembe.

A két módszer különbségét mutatja be a 3-4. ábra egy rádió adó-vevő készülék FMEA analízis számára történő strukturálásán keresztül.



3-4. ábra: Rádiókészülék FMEA felbontása strukturálisan és funkcionálisan

A harmadik modell a rendszerben lévő redundancia meghatározására szolgál, neve megbízhatósági blokkdiagram. Ez esetben az információ feldolgozását ábrázoljuk soros-párhuzamos kapcsolatok segítségével.

Az FMEA analízis eredményét többnyire táblázatos formában, leíró stílusban rögzítik, az alábbi struktúrában:

- a vizsgált komponens / alrendszer megadása,
- a vizsgált komponens vagy alrendszer célja, funkciója,
- a vizsgált meghibásodási mód,
- a meghibásodási mód magasabb egységre (alrendszer vagy rendszer) gyakorolt hatása, illetve detektálási módja.

3.6.3 Kritikusság vizsgálata

Az FMEA módszer, ahogy azt a fejezet bevezetőjében említettük, kiegészíthető a felfedett meghibásodási hatások (illetve az őket kiváltó meghibásodási módok) kategorizálásával, sorrendbe állításával. Ezzel a kiegészítéssel jobban ráirányítható a figyelem a gyakran bekövetkező, nagy fontosságú meghibásodásokra.

A kritikusság vizsgálatánál iparáganként más és más alapmódszert favorizálnak. A következőkben ezek közül mutatunk be kettő, a munka területéhez kapcsolódót.

Kockázat Prioritási Szám (Risk Priority Number - RPN) használata: A módszer elsősorban automatizálási területen használatos. Alapgondolata az egyes meghibásodási módok sorba állítása három kritérium szerint, amelyek a következők: fellépési gyakoriság, a következmény súlyossága és a meghibásodás detektálhatósága. A három kritérium szerinti rangsor alapján minden meghibásodási módhoz hozzárendelhető egy RPN szám:

RPN=fellépési gyakoriság sorszáma * következmény súlyosságának sorszáma * meghibásodás detektálhatóságának sorszáma

Az így nyert RPN számok alapján a meghibásodási módok fontossági sorrendbe állíthatóak (bár meg kell jegyezni, hogy a módszer speciális esetekben torzít, és hátrább rangsorol nagyobb figyelmet érdemlő eseményeket).

A másik fontos kritikusság vizsgálati módszer, amit elsősorban a nukleáris és a repülőgép iparban használnak, az elem kritikusság szám alkalmazása. Minden egyes elemre képzik az alábbi számot:

$$C_r = \sum_{n=1}^{j} (\alpha \cdot \beta \cdot \lambda \cdot t) \qquad \textbf{(3-6.)}$$

ahol α a meghibásodási mód arányszáma, β a meghibásodási mód hatásának bekövetkezési valószínűsége, λ a komponens meghibásodási rátája, t a komponens működési ideje, j pedig a komponens meghibásodási módjainak száma. A kapott számok alapján az elemek kritikussági sorrendje képezhető. A kritikussági számok lehetőséget biztosítanak rendszerek összehasonlítására, valamint a rendszer megbízhatóságának növelése a kritikussági számok csökkentésével egyenértékű.

3.7 Az eseményfa analízis

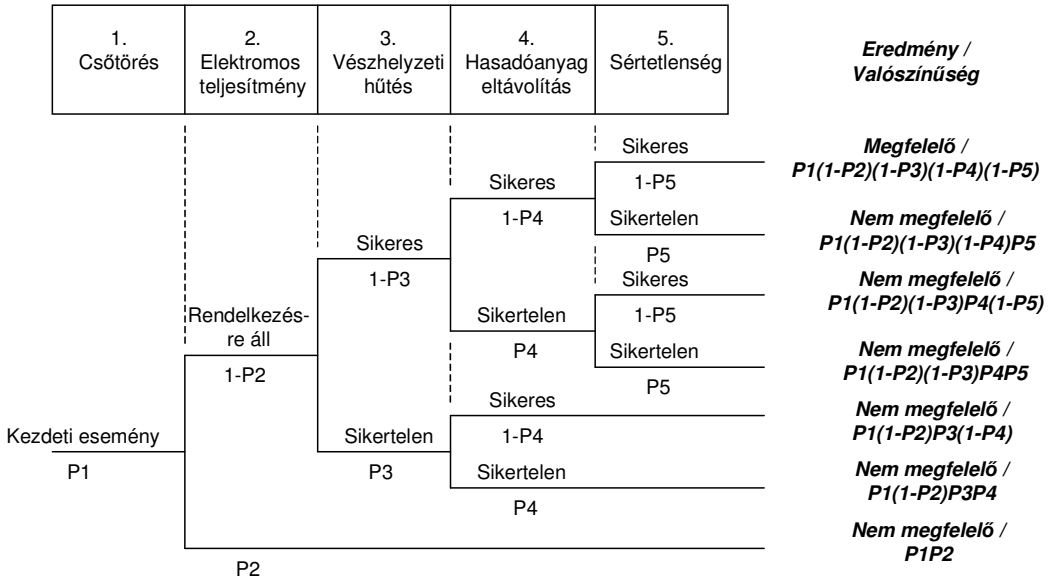
Az eseményfa (Eseményfa: Event Tree, ill. Eseményfa Analízis: Event Tree Analysis – ETA) bináris döntési fa, amelynek célja egy kezdeti esemény különböző feltételek melletti hatásainak vizsgálata [Leveson, 1995]. Az 1970-es évek elején kezdték alkalmazni olyan megbízhatóságanalízis-munkáknál, ahol a teljes rendszer modellezése a később bemutatandó hibafákkal már kezelhetetlenül nagy modellekhez vezetett volna. Valójában az eseményfa a közgazdaságtanban széleskörűen alkalmazott általános döntési fa adaptálása.

Az eseményfa kiindulási pontja egy kezdeti esemény (többnyire a vizsgált rendszer szempontjából külső esemény) bekövetkezése. Ezek után különböző kétállapotú feltételek (pl. meghibásodások) teljesülésének vagy nem teljesülésének hatására az eseményfa elágazik (illetve elágazhat). Amennyiben *n* esemény hatásait vizsgáljuk, ez 2^n ágat eredményezne az eseményfában. Az eseményfa méretének csökkentése érdekében a nem lehetséges, illetve fizikai tartalommal nem bíró ágakat a modellezés fázisában nem is hozzák létre. Az elágazásokat okozó események figyelembevételi sorrendje nem tetszőleges, célszerű a fizikai hatásokat követni a modellezésnél (pl. védelmi rendszerek a működésük sorrendjében). Következésképpen az eseményfa ott alkalmazható hatásosan, ahol az egyes figyelembe veendő események között sorrendiség állítható fel. Hasonlóan, az eseményfa alkalmazása nehézkes a sok egyenrangú eseményt tartalmazó rendszerekben.

Az egyes kialakuló ágakat (más néven vágatokat) következményük alapján kategóriákba sorolják (pl. biztonságos reakció vagy veszélyes állapot). Az egyes vágatok bekövetkezési valószínűsége a vágatot kialakító események bekövetkezési, illetve be nem következési valószínűségeinek szorzata (attól függően, hogy egy adott esemény bekövetkezése, vagy be nem következése szükséges a vágat kialakulásához). Az egyes vágatok bekövetkezése egymást kizáró esemény, így

amennyiben több vágat bekövetkezése is azonos következménnyel jár, a bekövetkezési valószínűségek összegezhetőek.

Eseményfa modellre mutat példát a 3-5. ábra. A példa [Leveson, 1995] alapján egy erőművi veszélyhelyzet kialakulását elemzi.



3-5. ábra: Példa eseményfa analízisre

Az eseményfa sok esemény figyelembevételére már nem praktikus. Éppen ezért alkalmazását hibafa-analízissel kombinálják. Ilyenkor az eseményfában figyelembe vett, elágazásokat okozó események különálló hibafák csúcseseményei. Ez a kombináció kiküszöböli az eseményfa korábban említett hátrányát is, miszerint az egyenrangú események modellezése a módszerrel nem célszerű – ilyenkor ezt a problémát a hibafa-analízis oldja meg.

3.8 A hibafa analízis

3.8.1 Bevezetés

A hibafa-analízis (Fault Tree Analysis – FTA) az egyik legszélesebb körben elterjedt megbízhatóság-analízis módszertan. Az 50-es években történt kifejlesztése óta az elméleti háttere egyre megalapozottabb, egyre gyorsabb és gyorsabb számítást lehetővé tévő algoritmusokat fejlesztenek a hibamodell feldolgozására, és a számítási elméletek alapján számtalan, a gyakorlatban jól használható algoritmus és szoftver csomag született [Brow, 1990], [Chunning és Dinghua, 1990], [Lee et. al.,1985], [Schneeweis, 1985], [Schneeweis, 1987], [Stecher, 1986], [Szabó, 1995]. E mellett számtalan alkalmazási javaslat és vizsgálati eredmény is napvilágot látott, pl. [Rastocny és Janota, 2000], [Bokor et. al., 1997], [Crosetti és Bruce,1970].

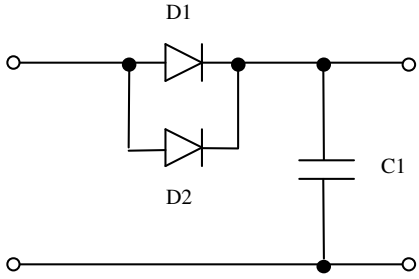
3.8.2 A hibafa-analízis módszertana

A hibafa-analízis (sok más analízishez hasonlóan) hibamodell-létrehozásból, majd a modell elemzéséből áll. A hibafa-modell felépítése top-down (felülről lefelé építkező) séma alapján történik. A kiindulási pont minden esetben egy rendszerszintű esemény (pl. “A rendszer működésképtelenné válik”), amelyet csúcseseménynek neveznek. A modell-alkotás célja annak modellezése, hogy a csúcsesemény mely, a rendszerben fellépő események (többnyire meghibásodások) hatásaként állhat elő. Azokat az eseményeket, amelyeket tovább már nem bontunk fel (elemi meghibásodások), alapeseményeknek nevezzük. Nagy rendszerek esetén célszerűtlen lenne a csúcseseményt egyből alapesemények kombinációjaként felírni. Ilyenkor un. közbenső eseményeket iktatunk a hibafába: ezek az események további felbontást igényelnek.

A hibafában az egyes események kapcsolatainak modellezésére az ÉS és a VAGY logikai kapukat alkalmazzák. Egyes analízis algoritmusok engedik használni a NOT műveletet (többnyire csak alapesemény szintjén), és a KvN (N-ből K) logikai kapcsolatot is. A hibafa felépítésekor az események okait az adott rendszer folyamatábráján visszafelé, az eseménytől az ok irányába haladva nyomozzuk ki (deduktív analízis). Minden egyes lépésben veszünk egy okozatot, és keresünk hozzá egy vagy több eseményt (kiváltó okot), amely lehet elemi esemény, vagy közbenső esemény, amelyet a későbbiekben tovább bontunk. (Megjegyzendő, hogy már a hibafa felépítése is igen hasznos segítséget nyújthat: rákényszeríti és rávezeti az analízist végző személyt, hogy vegye számba az összes eseményt, amelyek a csúcseseményhez vezethetnek.)

Az analízisnek több célja lehet:
1. Meghatározhatóak a csúcsesemény bekövetkezéséért felelős eseménykombinációk (minimális vágatok elemzése).
2. Meghatározható a csúcsesemény bekövetkezésének gyakorisága vagy valószínűsége (valószínűségi alapú elemzés).
3. Meghatározható a csúcsesemény bekövetkezési gyakoriságának vagy valószínűségének időfüggése (időfüggő elemzés).
4. Meghatározható a csúcsesemény bekövetkezési gyakoriságának vagy valószínűségének függése az egyes alapesemények értékeinek változásától (érzékenység vizsgálat).

Lássunk egy egyszerű példát a hibafa felépítésére. Példánkban egy soros diódás egyenirányító hibamodelljét alkotjuk meg. A kapcsolásban az egyenirányító dióda a meghibásodások elleni védelem miatt kettőzött. A kapcsolási rajzot a 3-6. ábra, míg a felépített hibafát a 3-5. ábra mutatja.



3-6. ábra: Diódás egyenirányító kapcsolási rajza

Első lépésként definiálnunk kell a csúcseseményt, melynek bekövetkezési valószínűségét szeretnénk számítani. Ez a mi esetünkben a következő esemény: “Az egyenirányító kimenetén nem jelenik meg az egyenirányított feszültség (vagy 0 V, vagy simítatlan egyenirányított feszültség jelenik meg) ”.

A modellünkben az alábbi meghibásodási típusokat vettük figyelembe (a figyelembe veendő hibatípusok függhetnek a definiált csúcseseménytől):

- Dx dióda szakadttá válik,
- Dx dióda zárlatossá válik,
- Cx kondenzátor üzemképtelen lesz (szakadt vagy zárlatos).

A meghibásodási típusok ismeretében definiálhatjuk azokat az elemi eseményeket, amelyeket a csúcsesemény szempontjából figyelembe kell venni (hibakatalógus). A hibakatalógust mutatja a 3-1. táblázat.

3-1. táblázat: Hibakatalógus

Az esemény sorszáma	*Megnevezés*	*Bekövetkezési valószínűség (fiktív érték)*
1.	D1 dióda szakadttá válik	$1.0*10^{-3}$
2.	D1 dióda zárlatos lesz	$1.5*10^{-6}$
3.	D2 dióda szakadttá válik	$1.0*10^{-3}$
4.	D2 dióda zárlatos lesz	$1.5*10^{-6}$
5.	C1 kondenzátor üzemképtelen lesz	$3.2*10^{-6}$

A következő lépésben felfedjük az egyes elemi eseményeknek a csúcsesemény bekövetkezésében játszott szerepét. A hibafa könnyebb felépítése és a könnyebb érthetőség miatt közbenső eseményeket is definiálunk.

A definiált közbenső események (ezek - eltérően az elemi eseményektől - tetszőlegesek lehetnek, illetve alkalmazásuktól el is lehet tekinteni) a következők:

- “A diódákon keresztül nem folyik áram”, illetve
- “A diódák rövidzárként viselkednek”.

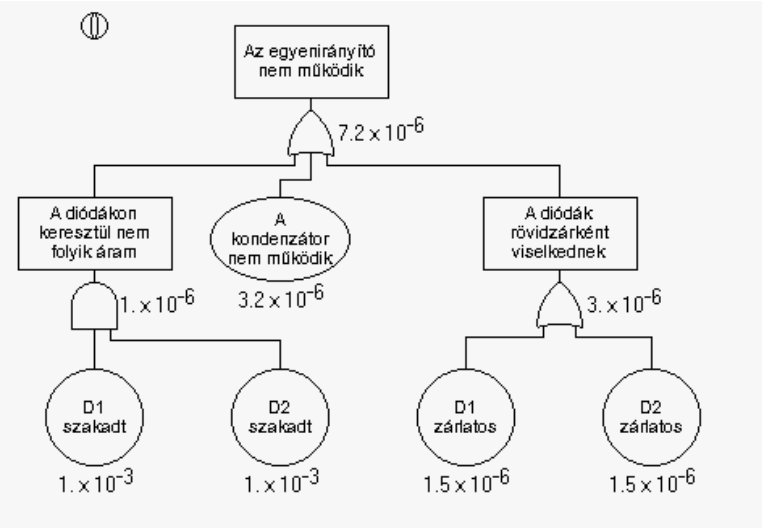
A csúcsesemény (“Az egyenirányító kimenetén nem jelenik meg az egyenirányított feszültség”) az alábbi három esemény valamelyikének bekövetkezésekor következik be: “A diódák rövidzárként viselkednek”, “A diódákon keresztül nem folyik áram”, illetve “C1 kondenzátor üzemképtelen lesz”. A csúcsesemény a három esemény VAGY kapcsolata.

Látható, hogy ezek közül csak egy az elemi esemény, míg a további kettő közbenső esemény, melyeket tovább tudunk bontani az alábbiak szerint:

“A diódák rövidzárként viselkednek” esemény akkor következik be, ha a két dióda közül legalább az egyik zárlatos, így ez az 3-1. táblázat 2-es és 4-es elemi eseményeinek VAGY kapcsolata.

“A diódákon keresztül nem folyik áram” esemény akkor következik be, ha mindkét dióda szakadttá válik, így ez az 3-1. táblázat 1-es és 3-as elemi eseményeinek ÉS

kapcsolata. A felépített hibafát (az alapesemények felvett és a csúcsesemény, illetve közbenső események kiszámolt bekövetkezési valószínűségével) a 3-7. ábra mutatja.



3-7. ábra: Diódás egyenirányító hibafa-struktúrája

A 3-2. táblázat a példahálózat minimális vágatait sorolja fel.

3-2. táblázat: Diódás egyenirányító minimális vágatai

Sorszám	Valószínűség	1. elemi esemény	2. elemi esemény
1.	3.2 e-6	A kondenzátor nem működik	-----
2.	1.5 e-6	D1 zárlatos	-----
3.	1.5 e-6	D2 zárlatos	-----
4.	1 e-6	D1 szakadt	D2 szakadt

3.8.3 A hibafa-analízis matematikai háttere

A következőkben a hibafa-analízis matematikai hátterét tekintjük át [Lee et. al.,1985], [Leitch, 1995] és [Rastocny és Janota, 2000] alapján.

3.8.3.1 Logikai kapcsolatok az alapesemények és a csúcsesemény között

A rendszert binárisan modellezzük az alábbiak szerint: A hibafában *n* alapesemény található, u_i jelöli az *i.* alapesemény állapotát, i=1,2, … *n*.

$$\begin{aligned} &u_i = 1, && \textit{ha az alapesemény bekövetkezett},\\ &u_i = 0, && \textit{ha az alapesemény nem következett be},\\ &\Psi(U) = 1, && \textit{ha a csúcsesemény bekövetkezett},\\ &\Psi(U) = 0, && \textit{ha a csúcsesemény nem következett be.}\\ &U = (u_1, u_2, \ldots, u_n) \end{aligned} \qquad \textbf{(3-7.)}$$

$\Psi(U)$ az alapesemények és a csúcsesemény logikai kapcsolatát modellező függvény, amelyre a következő állítások igazak:

$$\begin{aligned} &\Psi(U)=1, \qquad ha\, U=(1,1,\ldots,1),\\ &\Psi(U)=0, \qquad ha\, U=(0,0,\ldots,0)\, és\\ &\Psi(U)\geq\Psi(X) \quad ha\, u_i \geq x_i\, az\, összes\, elemre \end{aligned} \qquad \textbf{(3-8.)}$$

Amennyiben $\Psi(U)$-t ki szeretnénk fejezni, ezt megtehetjük a minimális vágatokra bontott rendszerben a minimális vágatok logikai függvényét leíró $R_j(U)$ függvények segítségével, ahol *j*=1,2,…*m*, *m* a minimális vágatok száma.

$$\Psi(U)=\bigcup_{j=1}^{m} R_j(U) \qquad \textbf{(3-9.)}$$

3.8.3.2 A minimális vágatok meghatározása

A minimális vágatok meghatározása a csúcsesemény irányából történik az alapesemények irányába. A kiindulás egy, a csúcsesemény azonosítóját tartalmazó halmaz. Amennyiben a csúcsesemény közbenső- vagy alapesemények ÉS kapcsolata, a halmazban a csúcsesemény azonosítóját cserélni kell az ÉS kapcsolatban részt vevő események azonosítóira (a halmaz elemeinek a száma növekszik). Amennyiben a csúcsesemény közbenső- vagy alapesemények VAGY kapcsolata, az eddigi (a vizsgált csúcseseményt tartalmazó) halmaz helyett annyi számú új halmazt kell létrehozni, ahány esemény vesz részt a VAGY kapcsolatban, és minden egyes új halmazban a csúcsesemény azonosítóját rendre az egyik, VAGY kapcsolatban részt vevő esemény azonosítójára kell cserélni.

Amennyiben az első lépés elvégzése után a halmaz (vagy halmazok) alapesemény-azonosítókon kívül tartalmaznak még közbenső esemény azonosítókat is, az ezekhez tartozó közbenső eseményeket a csúcseseménynél megismert algoritmus szerint tovább kell bontani. A folyamatot addig szükséges ismételni, amíg a halmazok még tartalmaznak közbenső esemény azonosítókat is.

A folyamat eredményeképpen előállnak a vágatok (az eseménytér olyan részhalmazai, amelyekben lévő alapesemények egyidejű bekövetkezésekor a csúcsesemény is bekövetkezik). Ezek a vágatok azonban nem feltétlenül minimális vágatok: a minimális vágat az eseménytér olyan részhalmaza, amelyben lévő alapesemények egyidejű bekövetkezésekor a csúcsesemény is bekövetkezek, de ha az események közül akár csak egy nem következik be, már maga a csúcsesemény sem következik be. A minimális vágatok vágatokból történő előállításához további két lépést kell elvégezni: az első lépésben az esetlegesen jelen lévő ismételt alapeseményeket kell redukálni. Az ismételt alapesemények egy vágatban egynél többször szereplő alapesemények, amelyeket a Boole-algebra szabályai szerint egyszeres alapeseménnyel helyettesítünk. Ezek után második lépésként rendre meg kell vizsgálnunk, hogy az egyes vágatok nem részhalmazai-e más vágatoknak. Amennyiben igen, akkor megállapíthatjuk, hogy az a vágat, amelyiknek van vágat részhalmaza, nem minimális vágat.

A fenti algoritmus ÉS és VAGY kapcsolatokat képes értelmezni az események bekövetkezései között – ez a két alapkapuja a hibafa-analízisnek. Az N-ből K kapuk (pl. 3-ból 2) kezeléséhez a kaput először helyettesíteni kell az ekvivalens funkciót megvalósító, ÉS és VAGY kapukat tartalmazó logikával, majd ez után a fentebb bemutatott módon elvégezhető az analízis. Az algoritmus NEM (NOT vagy negálás) kaput csak közvetlenül az alapesemény után tud értelmezni: ilyenkor a vágatba az alapesemény negáltja kerül.

A minimális vágatok önmagukban is nagyon fontos információt hordoznak: segítségükkel megállapítható, hogy egyidejűleg minimálisan hány meghibásodás fellépése szükséges a rendszerszintű esemény (rendszerhiba) bekövetkezéséhez. Sok esetben követelmény pl. az egyszeres hibatűrés teljesítése, ami azt írja elő, hogy bármely, a rendszerben fellépő egyedi meghibásodás ne vezessen rendszerhibához. Ennek a követelménynek a teljesülése könnyen ellenőrizhető a minimális vágatok segítségével. Ha az összes minimális vágat kettő, vagy annál több eseményt tartalmaz (másod- vagy magasabb rendű minimális vágatok), akkor az egyszeres hibatűrés feltétele teljesül.

3.8.3.3 A csúcsesemény bekövetkezési valószínűségének számítása minimális vágatok alapján

A csúcsesemény bekövetkezési valószínűsége kifejezhető a minimális vágatok segítségével:

$$P(\Psi(U)) = P\left(\bigcup_{j=1}^{m} R_j(U)\right) \qquad \textbf{(3-10.)}$$

Az unió felbontható az alábbi összefüggés segítségével:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B) \qquad \textbf{(3-11.)}$$

A 3-11. képletet alkalmazva a minimális vágatok esetére:

$$P\left(\bigcup_{j=1}^{m} R_j\right) = \sum_{j=1}^{m} P(R_j) - \sum_{j<k} P(R_j \cap R_k) + \sum_{j<k<l} P(R_j \cap R_k \cap R_l) \mp \ldots \pm P\left(\bigcap_{j=1}^{m} R_j\right) \qquad \textbf{(3-12.)}$$

A 3-12. képlet pontosan megadja a csúcsesemény bekövetkezési valószínűségét. Az egyes tagok számításával rendre alul vagy felülbecsüljük a pontos eredményt, így az egyes tagok alapján a számítás aktuális állásához tartozó pontosság megadható.

Számítástechnikailag hatékonyabb algoritmust kapunk a Boole-algebra tulajdonságainak alkalmazásával. E szerint:

$$\bigcup_{j=1}^{m} R_j = R_1 \cup (\overline{R}_1 R_2) \cup (\overline{R}_1 \overline{R}_2 R_3) \cup \ldots \cup (\overline{R}_1 \overline{R}_2 \overline{R}_3 \ldots \overline{R}_{m-1} R_m) \qquad \textbf{(3-13.)}$$

Mivel a jobb oldal tagjai egymást kizáró események, a bekövetkezési valószínűségük összeadható.

$$P\left(\bigcup_{j=1}^{m} R_j\right) = P(R_1) + P(\overline{R}_1 R_2) + P(\overline{R}_1 \overline{R}_2 R_3) + \ldots + P(\overline{R}_1 \overline{R}_2 \overline{R}_3 \ldots \overline{R}_{m-1} R_m) \qquad \textbf{(3-14.)}$$

3.8.3.4 A csúcsesemény bekövetkezési valószínűségének számítása minimális vágatok nélkül

Az olyan rendszerek hibafa-modelljénél, amelyek szeparált redundanciákat tartalmaznak (vagyis nem tartalmaznak ismételt alapeseményeket), lehetséges a csúcsesemény bekövetkezési valószínűségének közvetlen számítása az alapesemények bekövetkezési valószínűségéből. Ebben az esetben a számítás alulról felfelé elv alapján az alapeseményekkel közvetlenül kapcsolatban lévő transzfer események bekövetkezési valószínűségének számításával indul, és folytatódik egészen a csúcseseményig.

A számításokhoz használt képletek a következők.

Az esemény bekövetkezési valószínűsége, ha *n* esemény ÉS kapcsolataként került felírásra:

$$P_{ki} = \prod_{i=1}^{n} P_i \qquad \textbf{(3-15.)}$$

Az esemény bekövetkezési valószínűsége, ha *n* esemény VAGY kapcsolataként került felírásra:

$$P_{ki} = 1 - \prod_{i=1}^{n} \left(1 - P_i\right) \qquad \textbf{(3-16.)}$$

A módszer előnye az egyszerűsége, hátránya a korlátozott alkalmazhatóság.

3.8.3.5 Időfüggő analízisek

Az időfüggő analízisek célja a csúcsesemény bekövetkezési gyakoriság vagy valószínűség időbeli változásának meghatározása. Az időfüggő analízisek az alapesemények szintjén a bekövetkezési valószínűség időbeli változását leíró modellek alkalmazását igénylik. Az időfüggés számítása a korábban bemutatott módszereken alapul, azok megadott időpontokra való ismétlésével.

3.8.3.6 Érzékenységvizsgálatok

Az érzékenységvizsgálatok célja annak megállapítása, hogy mennyiben függ a rendszer megbízhatósága egy komponens vagy komponens típus megbízhatóságától. Három szokásos paramétert mutatunk be:

1. Kockázatnövelési tényező. A tényező azt mutatja meg, hányszorosára nő a csúcsesemény bekövetkezési valószínűsége, ha a vizsgált alapesemény (vagy alapesemény-típus) bekövetkezési valószínűsége 1-re nő (az alkatrész hibás lesz). A tényező azt mutatja meg, hogy egy komponens esetlegesen bekövetkező meghibásodása (bármekkora valószínűséggel is történik meg) mennyire kritikus, illetve érdemes-e az adott komponensnél redundanciát alkalmazni.
2. Kockázatcsökkentési tényező: A tényező azt mutatja meg, hányadrészére csökken a csúcsesemény bekövetkezési valószínűsége, ha a vizsgált alapesemény (vagy alapesemény-típus) bekövetkezési valószínűsége 0 lenne (tökéletes alkatrész). A tényező azt mutatja meg, hogy a komponenst érdemes-e nagyobb megbízhatóságú másik komponenssel helyettesíteni.
3. Kockázatváltozási tényező: A tényező két csúcsesemény bekövetkezési valószínűség hányadosa. Az első a vizsgált alapesemény bekövetkezési

valószínűségének tízszeres értékénél kerül számításra, míg a másik a vizsgált alapesemény bekövetkezési valószínűségének tizedénél. A tényező azt mutatja meg, mennyire érdemes egy komponenst jobbra cserélni, illetve a rendszer-megbízhatóság növelése érdekében melyik komponenseket érdemes javítani.

3.9 A Markov analízis (Automata kockázati analízis)

Automatán a rendszer állapotokat és a közöttük lévő átmeneteket értjük. Az átmeneteknek feltételeik vannak és egy kimeneti akció tartozik hozzájuk. Ha egy feltétel egy átmeneten igaz lesz, és a rendszer az ehhez tartozó állapotban van, akkor a rendszer az új állapotba kerül és végrehajtja a kimeneti műveletet.

Az analízis célja: A rendszer lehetséges állapotainak, illetve az állapotátmeneteknek a leírása, majd analízise. Egy adott állapotban (vagy állapotcsoportban) tartózkodás valószínűségének meghatározása.

Az analízis végrehajtási lépései:

1. A rendszer állapotterének meghatározása,
2. A vizsgálni kívánt állapotok meghatározása (kvázi csúcsesemény),
3. Az egyes állapotok közötti átmenetek definiálása (egy adott állapotból mely állapotokba lehet eljutni),
4. Az állapotátmenetek gyakoriságának meghatározása,
5. A rendszer analízise.

A módszer előnye:

- A rendszer dinamikus viselkedése is jól leírható vele.

A módszer hátránya:

- Nagyobb rendszerek lehetséges állapotainak száma igen nagy, a modell felépítése szinte lehetetlen.
- A vizsgált veszélyhelyzethez igen sok rendszerállapot tartozhat.

A 3-8. ábrán két komponensű meleg- és hidegtartalékolt rendszer Markov-modellje látható.

Állapotok: 1. Mindkét komponens jó, 2. Egy komponens jó, 3. Mindkét komponens hibás.
Jelölések: λ: az elemek meghibásodási rátája, μ: javítási ráta

3-8. ábra: Példa melegtartalékolt és hidegtartalékolt rendszer állapot diagramjára.

4. A HIBA-ADAPTÍV FUNKCIÓK ÉS ANALÍZISÜK

4.1 Bevezetés

A hibatűrési igényű vezérlőrendszerekben, különösen a biztonságorientált ipari rendszereknél a rendelkezésre állás vagy működőképesség szintjének növelése az egyik fontos cél. Ahogy a korábbiakban már utaltunk rá, a rendelkezésre állás és a működőképesség is mérhető valószínűségként, és sok esetben már a követelmények meghatározásánál specifikálják azokat a megbízhatóságra vonatkozó valószínűségi határértékeket, amelyeket a rendszernek teljesítenie kell. Ezek a határértékek korábbi munkákból levont tapasztalatokon alapulnak vagy hatóság által előírt értékek.

A megbízhatóság növelésének általános technikái a nagyobb rendelkezésre állású komponensek alkalmazása (safe-life technika), nagyobb fokú redundancia alkalmazása, a tesztelési periódusidő csökkentése stb. A korszerű, számítógép-alapú irányítási rendszerekben alkalmazható módszer a fenti módszerek helyett vagy mellett a hiba-adaptivitás. A hiba-adaptivitás ebben az összefüggésben a berendezés azon képességét jelenti, hogy képes az általa végrehajtott funkció megváltoztatására (átkonfigurálására) a rendszerben jelen lévő, detektált hibák hatására. Ennek a hiba-adaptív viselkedésmódnak az előfeltétele a hibák (vagy azok egy részének) detektálása [Soumelidis et. al., 1994], [Bokor et. al., 1991] , és az így nyert információ eljuttatása a feldolgozó helyekre. Általános elv szerint a hibadetektálás eredményeképpen jelentkező többletinformáció (a jel detektáltan hibás vagy nem) bináris státuszinformációként hozzárendelhető magához a jelhez.

Az adaptív komponensek a dinamikus rendszerkomponensek családjához tartoznak, és speciális kezelést igényelnek a megbízhatóság-analízis (pl. hibafa-analízis) számítások során. Az adaptív komponenseket tartalmazó rendszer analízise céljára a hibafa-analízis eljárást választottuk, részben azért, mert eljárásmódszertana jól kidolgozott, pl. [Lee et. al.,1985], [Stecher, 1986], és a módszertani elvek támogatására sok hatékony szoftver-eszköz létezik, pl. [Heger et. al., 1995], [Patterson-Hine és Koen, 1989], részben pedig azért, mert napjainkban a megbízhatóság-elmélet új kutatási irányainak egyike éppen a különböző dinamikus viselkedések (pl. számítógép újraindulás, hideg- és melegtartalékok esete) modellezése és analízise [Dugan et. al., 1990], és a kutatók nagy része is ezt a módszert favorizálja - esetenként Markov-láncokkal kombinálva. A Markov-láncokat alkalmazó modellek nehezen felépíthetőek, de jól analizálhatóak – ezért esetenként a hibafa-modellt (amely könnyen felépíthető és szemléletes rendszerreprezentáció) Markov-láncokká konvertálják, és úgy analizálják.

A hagyományos hibafa-analízis technika kétállapotú hibamodellel dolgozik: a meghibásodás által kiváltott esemény vagy bekövetkezik, vagy nem. Amikor a hiba-adaptív viselkedésmódot modellezzük, a kétállapotú modell már nem megfelelő, mivel az eseménynek három lehetséges állapota van: nem következett be; bekövetkezett, de erről nincsen biztos információnk (nem került detektálásra); illetve bekövetkezett, és a bekövetkezésről biztos információval rendelkezünk (detektálódott).

A három állapot kezelhető a hagyományos eszközökkel is, különválasztva a detektált és a nem detektált meghibásodásokat és az adaptivitást a klasszikus AND és OR kapukkal modellezve, de ezek a módszerek több hátránnyal is járnak:

- Sajnos az átkonfigurálódó logikák egy részének modellezéséhez elengedhetetlen NOT és XOR kapuk használata is – ezek a kapuk azonban a hibafa-analízis alap-elemkészletében nincsenek benne, és a legtöbb ismert módszer nem is képes velük számolni.
- A detektál és nem detektált meghibásodások (illetve meghibásodási módok), két különálló eseményként való kezelése azzal a hátránnyal is jár, hogy a komplex alapesemény modellek, amelyek a meghibásodási valószínűség időfüggését írják le (lásd következő fejezet) nem használhatóak [Gáspár és Szabó, 1998b].

A valós életben sok olyan rendszer található, amelyek viselkedése (meghibásodási módjai, vagy pontosabban rendszerfunkció-degradációi) nem írhatóak le kétállapotú modellekkel. Az ilyen rendszerek analíziséhez többállapotú hibafa-analízis technikát fejlesztettek ki, amely általánosságban minden eseménynél *n* lehetséges degradációs állapotot kezel, és a degradációs állapotok számának azonossága sem követelmény [Aven, 1985], [Caldarola, 1980], [Garribba et. al., 1985]. Ennek az analízistechnikának az alkalmazása általános esetben meglehetősen bonyolult, ami komoly alkalmazási hátrányt jelent (ellentétben a kétállapotú hibafa-analízissel, amely a leggyakrabban alkalmazott megbízhatóság-analízis technika, a többállapotú hibafa-analízist, illetve a többállapotú rendszeranalízist nagyon ritkán alkalmazzák). Ugyanakkor bizonyos szűkítő peremfeltételekkel alkalmazva a módszert, megoldást jelenthet az adaptivitás modellezésében is.

Utolsó módszerként az általunk speciálisan a bemutatott problémakörhöz kifejlesztett, zárt képletek technikájának nevezett analízis-eljárást mutatjuk be, amely a vezérlőrendszerek egy osztályára (elválasztott redundanciát tartalmazó ipari biztonságkritikus rendszerek) a többi módszernél gyorsabb számítást biztosít.

Noha a szakirodalomban található néhány, a dinamikus rendszerviselkedés modellezésével foglalkozó cikk, pl. [Doyle et. al., 1995], [Dugan et. al., 1990], [Gulati és Dugan, 1997], a hiba-adaptív viselkedésmódra vonatkozó analízis-eljárásokkal nem lehet találkozni. A téma elméleti kérdéseivel kapcsolatos cikkeink, [Gáspár és Szabó, 1998a], [Szabó és Gáspár, 1999a], [Szabó és Gáspár, 1999b] ennek következtében az általános hibafa-analízis kutatásokon alapulnak (lásd 2. fejezet), míg a gyakorlati alkalmazás lehetőségét az elmélet alapján tárgyaló cikkeink [Szabó és Gáspár, 1998a], [Szabó és Gáspár, 1998b] az elméleti eredményeinken.

4.2 Hiba-aktív és hiba-adaptív funkciók

A hibadetektálási információ birtokában hibaterjedést gátló mechanizmusok alkalmazhatóak a rendszerben, pl. a hibás jel előre definiált értékű jellel helyettesíthető. (E mellett természetesen vannak olyan megoldások, amelyek a hibaterjedés akadályozásához nem igénylik a detektálási információt, mint például a hagyományos szavazólogikák.)

Természetesen nem szükséges a fenti mechanizmusok alkalmazása a funkció-végrehajtás összes szintjén. Ennek következtében egy rendszer három típusú működési módot tartalmazhat a hiba-adaptív viselkedés szempontjából:

- Normál működési mód (normál logika): Ennél a módnál a funkcióvégrehajtásban nincs különbség detektáltan hibás és nem detektált hibájú vagy jó jellel való táplálás között (nincs átkonfigurálódás).
- Aktív működési mód (aktív logika): A funkcióvégrehajtás hasonló a normál működési módhoz, de a funkciót megvalósító egység figyeli a bemeneti jelek státuszait, és ha azok alapján helyesen nem hajtható végre a funkció (detektált hiba), ezt az információt hozzárendeli a kimeneti jelhez (nincs átkonfigurálódás, van aktív hibadetektálás).
- Adaptív működési mód (adaptív logika): A funkcióvégrehajtás mikéntje változik a bemenő jelekhez tartozó hibadetektálási információk alapján. Amennyiben a bemeneti jelek detektált hibás volta miatt további átkonfigurálódás már nem lehetséges, és az utoljára kiválasztott funkció sem hajtható végre helyesen, ezt az információt a logika hozzárendeli a kimeneti jelhez (van átkonfigurálódás, van aktív hibadetektálás).

A 4-1. táblázat egy 2v3 (3-ból 2) logika normál, aktív és adaptív megvalósítását mutatja. A három bemenet különböző állapotai mellett a kimenet állapotai sorolja fel a három működési típusnál.

4-1. táblázat: Normál, aktív és adaptív 2v3 funkció

Bem #1	Bem #2	Bem #3	Kim norm	Kim aktív	Kim adapt	Bem #1	Bem #2	Bem #3	Kim norm	Kim aktív	Kim adapt
N	N	N	N	N	N	D	D	U	U	D	U
N	N	D	N	N	N	D	U	N	U	U	N
N	N	U	N	N	N	D	U	D	U	D	U
N	D	N	N	N	N	D	U	U	U	U	U
N	D	D	U	D	N	U	N	N	N	N	N
N	D	U	U	U	N	U	N	D	U	U	N
N	U	N	N	N	N	U	N	U	U	U	U
N	U	D	U	U	N	U	D	N	U	U	N
N	U	U	U	U	U	U	D	D	U	D	U
D	N	N	N	N	N	U	D	U	U	U	U
D	N	D	U	D	N	U	U	N	U	U	U
D	N	U	U	U	N	U	U	D	U	U	U
D	D	N	U	D	N	U	U	U	U	U	U
D	D	D	U	D	D						

N: normál állapotú (nem hibás) jel,

D: detektáltan hibás jel,

U: nem detektáltan hibás jel.

4.3 Az adaptivitás kezelése a hibafa-analízisben

4.3.1 Makro-modellek alkalmazása

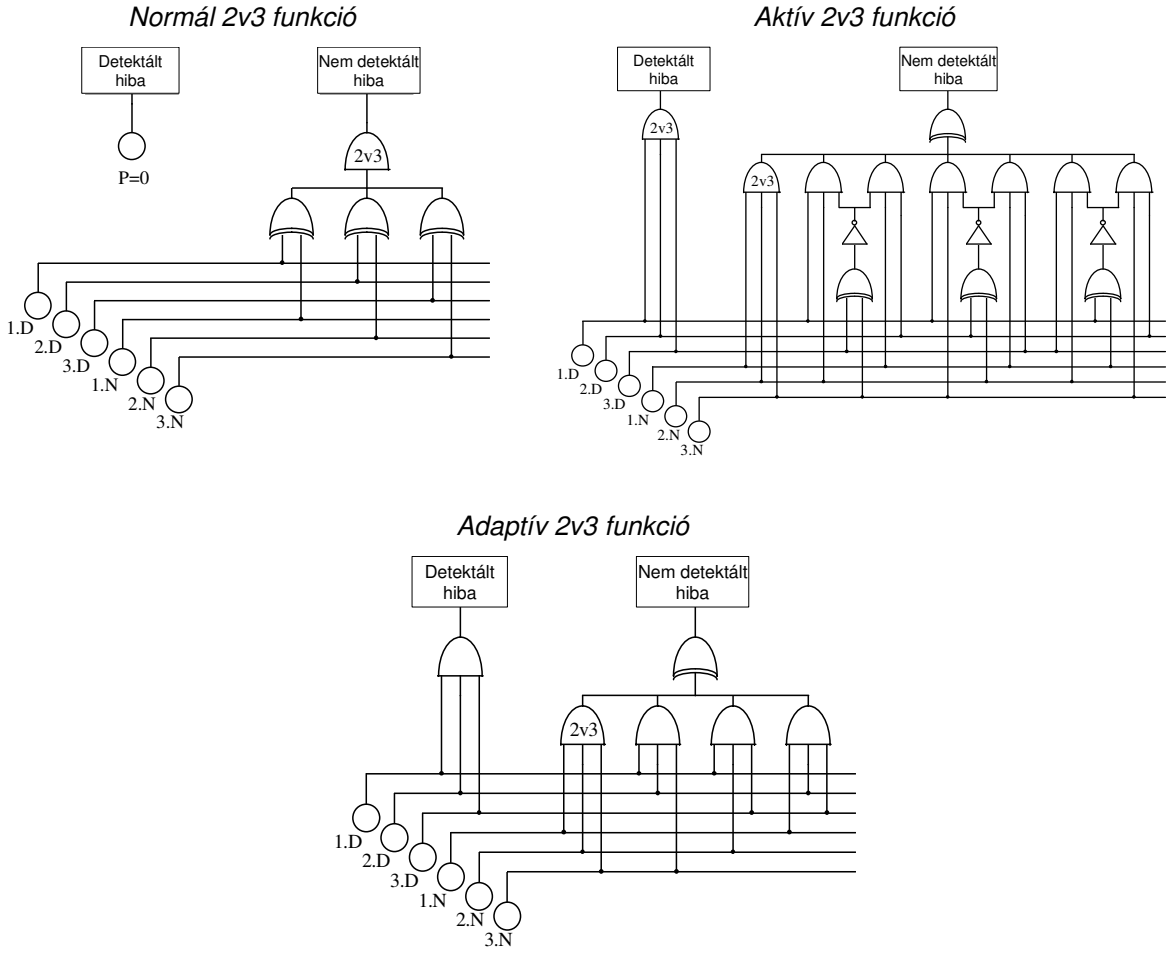
Amennyiben a hagyományos hibafa-analízis technikákkal, illetve módszertannal szeretnénk aktív és adaptív funkciókat is tartalmazó rendszert elemezni, külön kell választanunk az egyes meghibásodási alapesemények detektált és nem detektált részét, és mint egyedi alapeseményeket kell a rendszert modellező hibafába beépíteni. Az egyes aktív és adaptív funkciók meghibásodási, illetve hibaterjedési

szempontból vett modellje relatíve sok, hagyományos FTA kaput igényel, esetenként szükséges NOT és XOR kapuk alkalmazása is, ami a klasszikus kétállapotú hibafa-analízisnél nehézségeket okoz.

A modell-létrehozás megkönnyítése érdekében az egyes aktív és adaptív funkciók hibafa-modelljét előre definiálhatjuk, és a rendszeranalízis során az így létrehozott, un. makro-modellekkel dolgozhatunk. Ez az egyszerűsítés kiküszöböli a rendszer különböző helyein alkalmazott azonos típusú funkciók modellezésénél esetlegesen előforduló emberi hibákat. Ugyanakkor a makro-modellek nem jelentenek megoldást arra a problémára, hogy a hagyományos hibafa-analízisben az egy eseményhez tartozó detektált és nem detektált összetevők végig a teljes modellben külön-külön eseményként kerülnek modellezésre. Ráadásul a makro-modellek bonyolultsága miatt több, egymásra épülő szinten is aktív vagy adaptív funkciókat alkalmazó rendszer modellje kezelhetetlenül naggyá válik - illetve ha jelöléstechnikailag a makro-modellek alkalmazásával kezelhetővé is tehető, az analízis-eljárások végrehajtása okoz a modell nagy mérete miatt problémákat.

A makro modellek alkalmazásának másik problematikája az azonos típusú, de különböző bemenetszámmal dolgozó funkciók modellezése, amelyek külön-külön makro-modelleket igényelnek.

A 4-1. ábrán a 3-ból 2 szavazás (2v3) különböző eseteinek makro-modelles modellezését mutatjuk be.



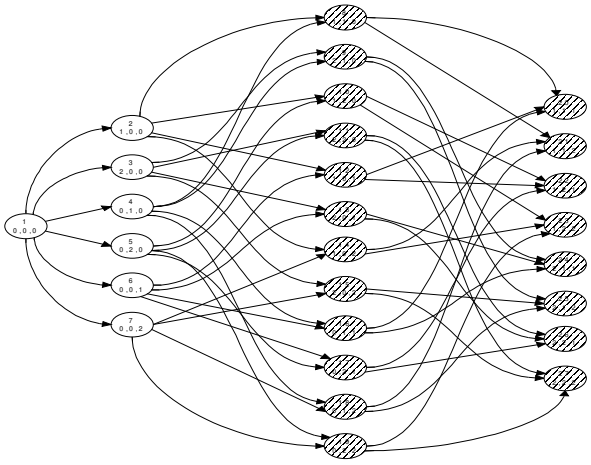
4-1. ábra: Makro modellek

4.3.2 Markov-lánccal történő modellezés

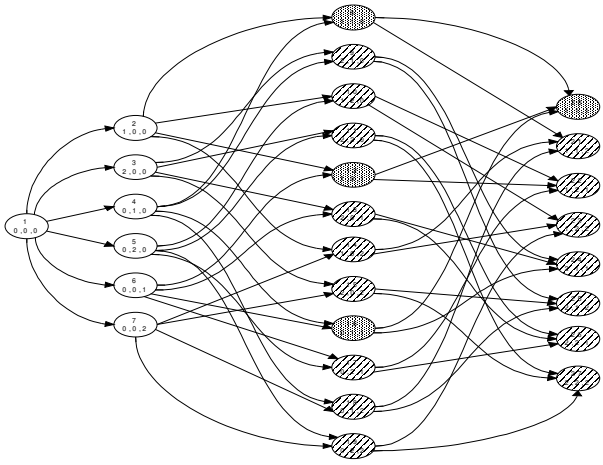
Amennyiben a meghibásodások Markov-tulajdonságokkal rendelkeznek, a hibafák (hibafa-modellek) Markov láncokká konvertálhatóak, és a hibaanalízis a láncok alapján végezhető el. Ennek a módszernek az előnye kettős: egyrészt kiküszöböli a hibafa-analízis azon hátrányát, hogy a tényleges analízis-eljárás során a rendszer dinamikus viselkedésének kezelése nehézkes vagy lehetetlen, másrészt kiküszöböli a dinamikus viselkedések kezelésére kiválóan alkalmas Markov-láncoknak azt a hátrányát, hogy a rendszermodell felépítése nagyon nehéz és átláthatatlan modellt eredményez. (A gyakorlatban a Markov modellezést néhány tíz, maximum néhány száz rendszerállapotig alkalmazzák, míg egy többszörös redundanciát tartalmazó rendszer meghibásodási szempontból vett állapottere ennél jóval nagyobb.)

A 4-2. (a és b) ábrán a 3-ból 2 szavazás (2v3) különböző eseteinek Markov láncokkal történő modellezését mutatjuk be. Minden egyes rendszerállapothoz egyedi állapotazonosítót és a funkció bemeneteinek státuszát leíró három számot (0: nem hibás, 1: detektált hiba, 2: nem detektált hiba) rendeltünk.

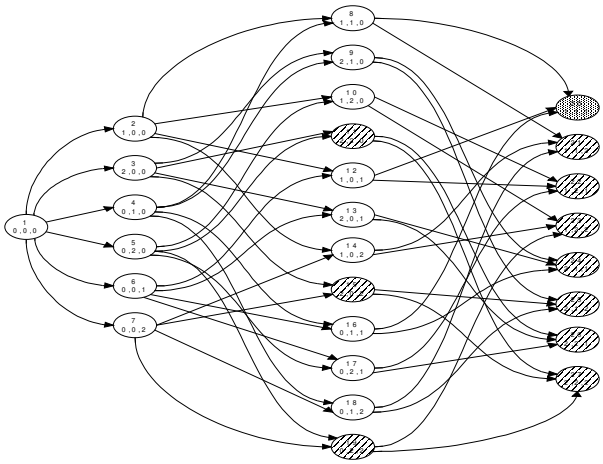
Normál 2v3 funkció



Aktív 2v3 funkció



Adaptív 2v3 funkció



4-2. (a) ábra: Markov modellek a 2v3 funkció modellezéséhez

Jelölések:

Hibamentes állapot

Detektáltan hibás állapot

Nem detektált hiba állapota

4-2. (b) ábra: Markov modellek a 2v3 funkció modellezéséhez - jelölésrendszer

A 4-2. (a) ábrában az állapotokat a jobb áttekinthetőség érdekében oszlopokba szerveztük. Az első oszlop (csak egy állapot) nem tartalmaz hibás bemeneti jelet, a második oszlopban található állapotok rendre egy hibás bemenetet tartalmaznak, a harmadik oszlop állapotai rendre két hibás bemenetet, míg a negyedik oszlop állapotaiban mindhárom bemenet detektált vagy nem detektált módon nem megfelelő.

Az ábrák csak (a funkció bemeneti jelein keresztül jelentkező) meghibásodások hatására fellépő állapotátmeneteket tartalmazzák, az átmeneti intenzitások a meghibásodási ráták. További állapotátmenetek vehetőek fel a tesztelések hatásának leírására (oszlopon belüli átmenetek nem detektált meghibásodást tartalmazó állapotból detektált hibát tartalmazó állapotba), valamint a javítások hatásának figyelembevételére (magasabb oszlopszámú oszlopból alacsonyabba való átmenet, visszalépés).

4.3.3 Többállapotú hibafa-analízis alkalmazása

A többállapotú hibafa-analízis újabb kezelési lehetőséget biztosít az adaptivitás kezelésére. Ennél a módszernél minden egyes eseménynek tetszőleges számú állapota lehet. Az egyes események közötti kapcsolat definiálására a klasszikus kapuk helyett igazságtáblák alkalmazhatóak, amelyek a kapcsolat összes lehetséges bemenő állapotához megadják a kimeneti állapotot. Amennyiben azonos funkciót megvalósító, de különböző bemenetszámmal rendelkező részt kell modellezni, ez csak különböző nagyságú (különböző számú sorral rendelkező) igazságtáblázatokkal tehető meg.

Esetünkben minden egyes eseménynek három lehetséges állapota van, a detektált hiba, a nem detektált hiba és a hibamentes állapot.

Általános logika esetére a kimeneti állapotok valószínűségét az α igazságtáblázat alapján Kai képletével [Kai, 1990] számolhatjuk:

$$p_i^{(t)} = \sum_{\substack{\ell_1,\ldots,\ell_k \\ \alpha_{\ell_1,\ldots,\ell_k}=t}} \prod_{j=1}^{k} p_{i_j}^{(\ell_j)}, \quad t = 0,1,2, \qquad \textbf{(4-1.)}$$

ahol p_i az i. eseményhez tartozó logika kimeneti valószínűségi vektora (3 állapotú), p_{i_j} a j. bemeneti valószínűségi vektor, (t) a kimenet, míg ℓ_k a k. bemenet állapota a három lehetséges közül.

A módszer hátránya az alkalmazáshoz szükséges igazságtáblázat méretében rejlik. A táblázat sorainak száma 3 állapot és *n* bemenet esetén 3^n. Ez három-négy bemenet esetén még tolerálható, de pl. 10 bemenet esetén (ilyen kapuk redundáns rendszerek esetén relatíve gyakran előfordulhatnak) már 59049 sorú táblázatot

igényel a képlet. Ennek a táblázatnak a létrehozása, tárolása és feldolgozása is nehézkes.

4.3.4 Zárt formulák alapján történő feldolgozás

Az igazságtábla alapján az egyes állapotok valószínűsége úgy is számolható, ha csak a detektált és nem detektált meghibásodások valószínűségét vesszük figyelembe. A következőekben zárt alakú képleteket adunk a hibafa-analízis két alapkapu-típusának három-három alváltozatához (ÉS kapu, VAGY kapu), valamint a leggyakoribb kiegészítő elemhez, a k/n kapuhoz. A képletek származtatásának bemutatásához az 1. mellékletben bemutatjuk a zárt alakú képletek létrehozását három bemenet esetére.

Az alább bemutatandó képletek általános kapukezelést biztosítanak, lényegesen megkönnyítve ezzel az analízist.

Az OR (VAGY) hibafa-kapu hibaesemény bekövetkezését jelzi a kimenetén, ha legalább egy bemeneti hibaeseménye bekövetkezett. Amennyiben a modellezett logika nem tesz különbséget detektált és nem detektált meghibásodás között a bemeneti oldalon, a kimeneti hibaesemény bekövetkezési valószínűsége az összes bemeneti meghibásodás VAGY kapcsolatához tartozó klasszikus képlet alapján számolható. Az így kiszámolt érték a következő logikai szint számára a nem detektált hiba valószínűsége.

Aktív logika esetén a kimeneten detektált és nem detektált meghibásodási valószínűséget is számolunk – természetesen ezek összege (mivel egymást kizáró események) meg kell hogy egyezzen a normál logika nem detektált kimeneti hibavalószínűségével.

Adaptív logika esetén olyan működésmódot feltételezünk, amely a hibafa-kapuval modellezett logikánál detektált bemeneti hiba esetén a hibás bemeneti jel vagy érték blokkolását (számításból való kivonását vagy előre definiált értékkel történő helyettesítését) vonja maga után. Ez a folyamat egészen addig folytatható, amíg jónak látott bemeneti értéket kap a logika. Természetesen nem detektált hiba esetén nincs működési mód váltás, hiszen a nem detektált hibát a logika nem látja.

A következő képletekben P_{Di} a detektált hiba bekövetkezési valószínűsége az i. bemeneten, míg P_{Ni} a nem detektált bemeneti hiba bekövetkezési valószínűsége.

$$P_{OR}^{Normal} = \sum_{m=1}^{N} \left[(-1)^{m+1} \sum_{\substack{P_{A\ell_j} \in \{P_{D\ell_j}, P_{N\ell_j}\} \\ \ell_j \in \{1 \ldots N\}: \ell_1 < \ldots < \ell_m}} \prod_{j=1}^{m} (P_{A\ell_j}) \right] \qquad \textbf{(4-2.)}$$

$$P_{Det,OR}^{Active} = \sum_{m=1}^{N} \left[(-1)^{m+1} \sum_{\ell_j \in \{1 \ldots N\}: \ell_1 < \ldots < \ell_m} \prod_{j=1}^{m} (P_{D\ell_j}) \right] \qquad \textbf{(4-3.)}$$

$$P_{UnDet,OR}^{Active} = \sum_{m=1}^{N} \left[(-1)^{m+1} \sum_{\substack{P_{A\ell_j} \in \{P_{N\ell_j}, P_{D\ell_j}\} \\ \ell_j \in \{1 \ldots N\}: \ell_1 < \ldots < \ell_m \\ \exists \ell_s \in \{\ell_1 \ldots \ell_m\}: P_{A\ell_s} = P_{N\ell_s}}} \prod_{j=1}^{m} (P_{A\ell_j}) \right] \quad \textbf{(4-4.)}$$

$$P_{Det,OR}^{Adaptive} = \prod_{j=1}^{N} (P_{Dj}) \quad \textbf{(4-5.)}$$

$$P_{UnDet,OR}^{Adaptive} = \sum_{m=1}^{N} \left[(-1)^{m+1} \sum_{\ell_j \in \{1 \ldots N\}: \ell_1 < \ldots < \ell_m} \prod_{j=1}^{m} (P_{N\ell_j}) \right] \quad \textbf{(4-6.)}$$

AND (ÉS) hibafa-kapu esetén az előző bekezdésben bemutatott, átkonfigurálás általi előny nem jelentkezik, hiszen a kimeneti hibaesemény csak az összes bemeneti hibaesemény együttes bekövetkezésekor következik be. Aktív modell esetén detektált kimeneti hibáról beszélhetünk, ha minden egyes bemenet hibás és detektáltak a hibák. Adaptív viselkedésmód definiálása itt szükségtelen, mivel a detektáltan hibás bemenet blokkolása a funkciót nem változtatja meg.

$$P_{AND}^{Normal} = \sum_{P_{Aj} \in \{P_{Nj}, P_{Dj}\}} \left[\prod_{j=1}^{N} (P_{Aj}) \right] \quad \textbf{(4-7.)}$$

$$P_{Det,AND}^{Active} = \prod_{j=1}^{N} (P_{Dj}) = P_{Det,AND}^{Adaptive} \quad \textbf{(4-8.)}$$

$$P_{Un\,det,AND}^{Active} = \sum_{\substack{P_{Aj} \in \{P_{Nj}, P_{Dj}\} \\ \exists j: P_{Aj} = P_{Nj}}} \left[\prod_{j=1}^{N} (P_{Aj}) \right] = P_{Un\,det,AND}^{Adaptive} \quad \textbf{(4-9.)}$$

A normál N-ből K (KvN) hibafa-kapu csak nem detektált kimeneti hibavalószínűséggel rendelkezik, a bemeneten a detektált és a nem detektált módon történő hibabekövetkezés között a logika számára nincs különbség. Az aktív logika detektált hibát produkál a kimeneten, ha a bemenetek közül K darab hibáját detektáltuk. Az adaptív logika átkonfigurálódik egy detektált hiba esetén Kv(N-1) logikává ($K \leq (N-1)$), illetve amennyiben az átkonfigurálódás már nem lehetséges, detektált hiba következik be a kimenetén.

$$P_{KvN}^{Norm} = \sum_{m=K}^{N}\left[(-1)^{m+K}\binom{m-1}{K-1}\sum_{\substack{P_{A\ell_j}\in\{P_{D\ell_j},P_{N\ell_j}\}\\ \ell_j\in\{1\ldots N\}:\,\ell_1<\ldots<\ell_m}}\prod_{j=1}^{m}(P_{A\ell_j})\right] \qquad \textbf{(4-10.)}$$

$$P_{Det,KvN}^{Active} = \sum_{m=K}^{N}\left[(-1)^{m+K}\binom{m-1}{K-1}\sum_{\ell_j\in\{1\ldots N\}:\,\ell_1<\ldots<\ell_m}\prod_{j=1}^{m}(P_{D\ell_j})\right] \qquad \textbf{(4-11.)}$$

$$P_{Undet,KvN}^{Active} = \sum_{m=K}^{N}\left[(-1)^{m+K}\binom{m-1}{K-1}\sum_{\substack{P_{A\ell_j}\in\{P_{N\ell_j},P_{D\ell_j}\}\\ \ell_j\in\{1\ldots N\}:\,\ell_1<\ldots<\ell_m\\ \exists\ell_s\in\{\ell_1\ldots\ell_m\}:\,P_{A\ell_s}=P_{N\ell_s}}}\prod_{j=1}^{m}(P_{A\ell_j})\right] \qquad \textbf{(4-12.)}$$

$$P_{Det,KvN}^{Adaptive} = \prod_{j=1}^{N}(P_{Dj}) \qquad \textbf{(4-13.)}$$

$$P_{Undet,KvN}^{Adaptive} = \sum_{m=K}^{N}\left[(-1)^{m+K}\binom{m-1}{K-1}\sum_{\ell_j\in\{1\ldots N\}:\,\ell_1<\ldots<\ell_m}\prod_{j=1}^{m}(P_{N\ell_j})\right] + \sum_{\substack{P_{A\ell_j}\in\{P_{N\ell_j},P_{D\ell_j}\}\\ \ell_j\in\{1\ldots N\}:\,\ell_1<\ldots<\ell_m\\ \exists\ell_s\in\{\ell_1\ldots\ell_k\}:\,P_{A\ell_s}=P_{N\ell_s}\\ \forall\ell_z\in\{\ell_1\ldots\ell_k\}:\,\ell_z\neq\ell_s:\,P_{A\ell_z}=P_{D\ell_z}}}\prod_{j=1}^{N}(P_{A\ell_j}) \qquad \textbf{(4-14.)}$$

I. Tézis: Megállapítottam, hogy a számítógép alapú vezérlőrendszerekben megvalósított hiba-aktív és hiba-adaptív funkciók megbízhatósági analízise hibafa-analízis módszerrel elvégezhető.

I.A: Megállapítottam, hogy a kérdéses funkciók modellezésére alkalmazható a makro-modellek módszere, a Markov-láncokká konvertálás, vagy az általános többállapotú hibafa-analízis.

I.B: Zárt alakú képleteket hoztam létre a hiba-aktív és hiba-adaptív funkciók optimális modellezéséhez és analíziséhez. Az általam létrehozott zárt alakú képletek szeparált redundanciákat tartalmazó vezérlőberendezések megbízhatósági analízisénél alkalmazhatóak.

5. A MEGBÍZHATÓSÁG IDŐFÜGGÉSE

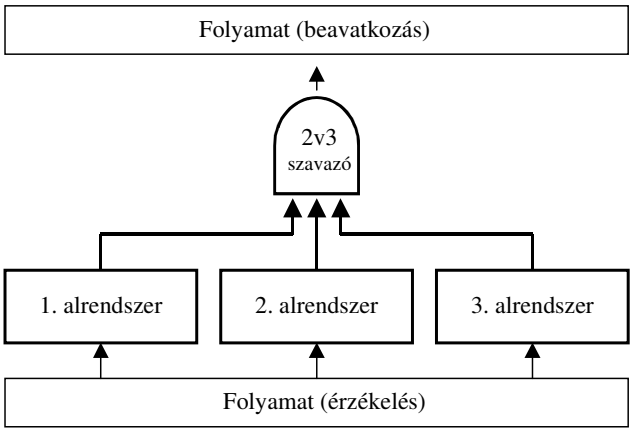
5.1 Bevezetés

A 3. fejezetben bemutatott megbízhatósági paraméterek jelentős része az idő függvényében változtatja az értékét. A változás oka két összetevőre bontható: egyrészt a meghibásodási ráta értéke is folyamatosan változik, másrészt a paraméterek nagy része a meghibásodási ráta exponenciális függvényeként írható le (egyes paraméterek, mint pl. a tesztelhetőség általában időinvariánsak). A rendszer-megbízhatóság időfüggésének vizsgálata az olyan helyeken kiemelten fontos, ahol előre definiált megbízhatósági limitértékeket kell teljesíteni, és ahol több részrendszer megbízhatóságának időfüggése külön-külön befolyásolható. Az időfüggés ismerete a megbízhatósági limitértékek eléréséhez szükséges tesztelési stratégiák meghatározása szempontjából is igen fontos [Bartha et. al., 2005], [Szabó et. al., 2008].

Példaként tekintsük a következő egyszerű rendszert (5-1. ábra): A nagy megbízhatóságot három független alrendszer szavatolja, amelyek a példában ideálisnak tekintet szavazólogikán (3/2-es szavazás) keresztül hajtják meg a beavatkozó pontot. (Az ideális szavazólogika nem teljesen utópisztikus fogalom: az elosztott szavazólogikák, pl. relés megvalósítással rendelkeznek ugyan számításba veendő meghibásodási valószínűséggel, de ez az egyes elosztott részeknél jelentkezik, amelyek meghibásodása a szavaztatni kívánt logikák meghibásodásával soros rendszereket alkot.)

A szavazólogika típusa miatt a beavatkozási ponton a működés elmaradás valószínűsége:

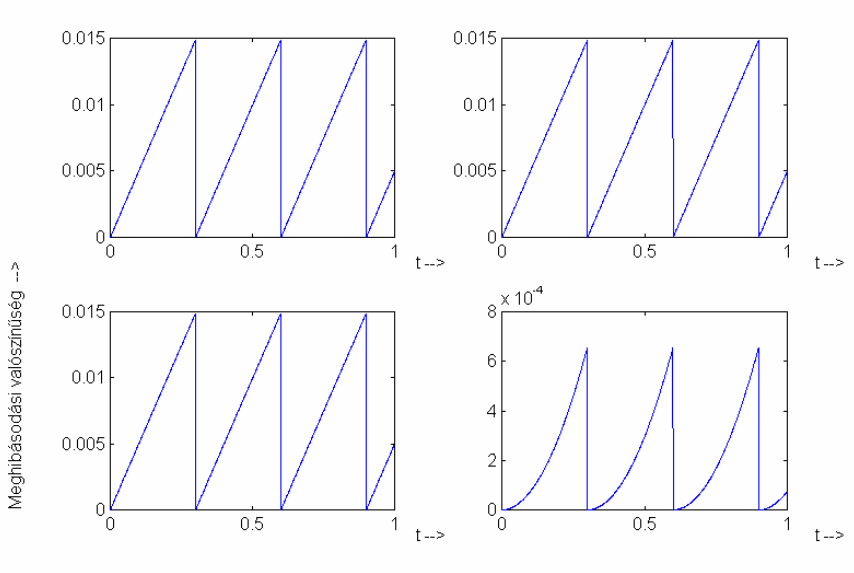
$$Q_B(t) = Q_1(t) \cdot Q_2(t) + Q_1(t) \cdot Q_3(t) + Q_2(t) \cdot Q_3(t) - 2 \cdot Q_1(t) \cdot Q_2(t) \cdot Q_3(t) \qquad \textbf{(5-1.)}$$



5-1. ábra: A példa rendszere

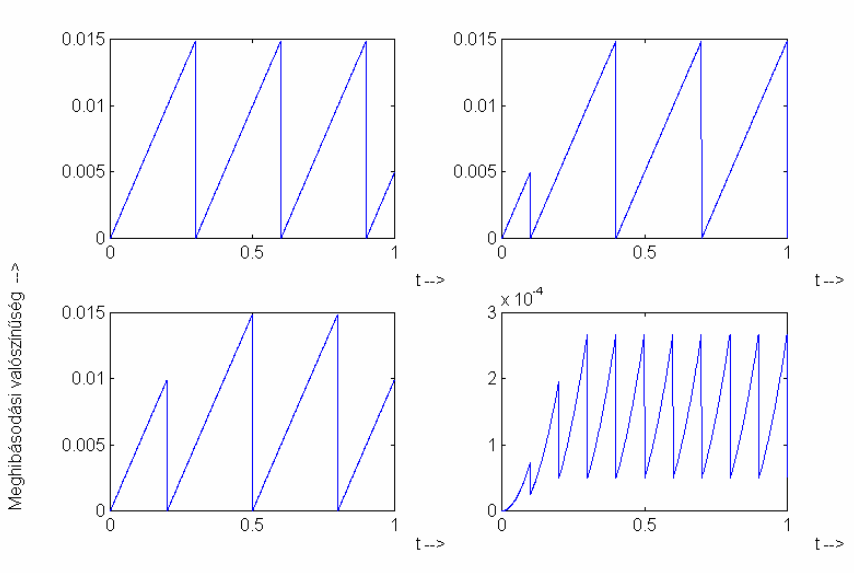
A példa három alrendszere periodikusan tesztelt, egyforma paraméterekkel rendelkező komponens (a komponens modelljét a későbbiekben részletesen bemutatjuk). Az első részben a periodikus tesztre mindhárom alkatrésznél egy időben (vagy azonnal egymás után, tehát lényegében egy időben) kerül sor, míg a

második részben a periodikus tesztek végrehajtását egyenletesen osztottuk el a tesztperiódusban. A rendszer eredő meghibásodási valószínűsége az 5-2. és 5-3. ábrákból láthatóan a második esetben lényegesen javul. Az ábrákon rendre az első, a második és a harmadik alrendszer, valamint a teljes rendszer meghibásodási valószínűségét ábrázoltuk. Az időtengely normalizált, a normálás alapja 1000 óra. Teszt intervallum: 300 óra. A második részben alkalmazott teszt-eltolások: 100 és 200 óra. Meghibásodási ráták egységesen 0,05 óra^{-1} .



5-2. ábra: A példarendszer eredményei tesztelési eltolás nélkül

A fentiek is mutatják, hogy nagy megbízhatóságú rendszerekben igen fontos az egyes komponensek paraméterei mellett a megbízhatóság időbeli változásának ismerete, a megfelelő valószínűségi modell alkalmazása az analízis során. A következőekben ezért bemutatjuk az időfüggést leíró általános, széles körben használt modelleket és ezek kiterjesztését az adaptív logikák analíziséhez (komplex modellek).



5-3. ábra: A példarendszer eredményei tesztelési eltolással

A továbbiakban a működőképességet és komplementer párját, a működésképtelenséget vizsgáljuk [Leitch, 1995], [RiskSpectrum b].

A működőképesség (reliability) R(t) időfüggvény egy komponens esetén, javítás nélkül az alábbi általános jellemzőkkel rendelkezik:

$$R(0)=1;\ \lim_{t\to\infty} R(t)=0 \qquad \textbf{(5-2.)}$$

és a függvény monoton csökkenő.

Sokszor szokás a meghibásodási valószínűséget vagy más néven működésképtelenséget Q(t) (unreliability) is kifejezni:

$$Q(0)=0;\ \lim_{t\to\infty} Q(t)=1 \qquad \textbf{(5-3.)}$$

és a függvény monoton növekvő.

$$Q(t)+R(t)=1 \qquad \textbf{(5-4.)}$$

A működésképtelenség értékének változási sebessége a meghibásodási sűrűségfüggvény:

$$f(t)=\frac{dQ(t)}{dt} \qquad \textbf{(5-5.)}$$

A meghibásodási gyakoriság (meghibásodási ráta):

$$\lambda(t)=\frac{f(t)}{R(t)}=\frac{\frac{dQ(t)}{dt}}{1-Q(t)} \qquad \textbf{(5-6.)}$$

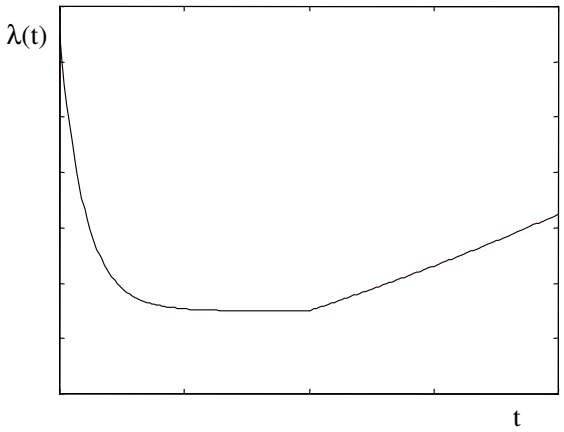
A jellemzők értékei lehetnek időben konstansok és lehetnek időfüggők, ilyenkor a konkrét időfüggvényen kívül az átlagérték és a maximális érték szolgálhat jellemzésül.

5.2 A meghibásodási ráta időbeli függése

A meghibásodási gyakoriság időbeli változását az 5-4. ábra mutatja. A grafikonban bemutatott időfüggvényt az 5-6. képlet alapján, nagy számú mintahalmazon elvégzett vizsgálattal lehet felvenni.

A görbe (alakja miatt kádgörbének is szokás nevezni) három jól elkülönülő szakaszra bontható. Az első szakaszt bejáratási szakasznak nevezik. Ebben a kezdeti szakaszban azok a példányok hibásodnak meg, amelyek valamilyen gyártási probléma miatt szerkezeti hibákat tartalmaztak. Ez a bejáratási szakasz igen rövid, általában a gyártás utáni ellenőrző tesztek alatt lezajlik. (Annak érdekében, hogy ez a szakasz valóban gyorsan elteljen, a gyártók speciális ellenőrző üzemeltetést alkalmaznak: megemelt hőmérsékleten üzemeltetik a berendezéseket. A megemelt hőmérséklet megnövekedett igénybevételt jelent a berendezésnek, így az gyorsabban öregszik. Ez az eljárást beégetésnek (burn-in) nevezik.) A következő szakaszban a meghibásodási gyakoriság közel konstans. Ez a normál működési tartomány szakasza, ilyenkor a véletlenszerű meghibásodások dominálnak. A

harmadik szakasz az öregedés fázisa, itt az anyagfáradások következtében a meghibásodási gyakoriság lassú, folyamatos növekedése figyelhető meg.



5-4 ábra: A meghibásodási ráta időfüggése (kádgörbe)

5.3 A rendelkezésre állás időfüggése

5.3.1 Bevezetés

A következőkben olyan egy komponensből álló rendszereket vizsgálunk meg, amelyekben megengedett a komponens cseréje vagy javítása a rendszer működőképességének fenntartása érdekében. Az alábbi modellek egy kivétellel a komponens meghibásodási gyakoriságának állandó értékét tételezik fel, vagyis a normál működtetési tartományra vonatkoznak. A komponens meghibásodásának időfüggését exponenciális függvénnyel közelítjük. Az exponenciális leírási mód a legszélesebb körben elterjedt, leggyakrabban alkalmazott leírási mód, de meg kell jegyezni, hogy léteznek más, bonyolultabb leírási modellek is, pl. Weibull fv.

A meghibásodási valószínűségi modellek feladata, hogy az alkatrész rendelkezésre nem állásának időbeli változását definiálják. Egyszerűbb valószínűségi számításoknál a bonyolultabb modellek helyett az időbeli átlagértéket is szokás használni.

5.3.2 Konstans rendelkezésre nem állású komponens

A legegyszerűbb modell esetén az alkatrész rendelkezésre állását konstansnak feltételezzük:

$$Q(t) = q \qquad \textbf{(5-7.)}$$

és így az átlagértékre az alábbi egyenlőséget kapjuk:

$$Q_{átlag} = q \qquad \textbf{(5-8.)}$$

A fenti modellel jellemzett alkatrész csak bekapcsoláskor hibásodhat meg, és a bekapcsoláskori meghibásodás valószínűsége *q*. Ha az elem bekapcsoláskor

meghibásodott, javítására lehetőség nincs, így bármely időpontban vizsgálva a rendelkezésre nem állását, a bekapcsoláskori *q* valószínűséget kapjuk (5-5. ábra).

Ilyen típusú valószínűségi modellel jellemezhetőek pl. az olyan szelepek, melyek két állapottal rendelkeznek és bekapcsoláskor váltanak állapotot. Amennyiben az állapotváltás már bekövetkezett, a szelep működésében nem léphet fel hiba, így csak a váltáskori (bekapcsolási) meghibásodást vesszük figyelembe. Hasonló módon modellezhetőek a hideg- és melegtartalékolt rendszerekben alkalmazott, nem ideális átkapcsolók.

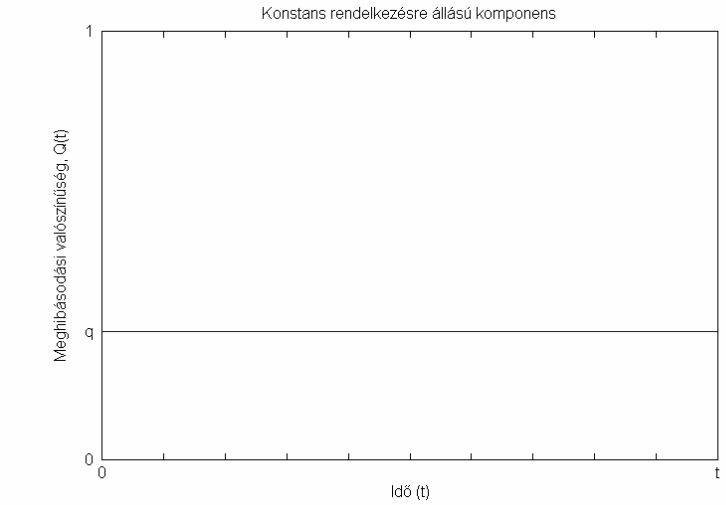
5.3.3 Nem javítható komponens

A leggyakoribb, nem javítható (vagy nem javított) komponens, konstans meghibásodási rátával. (A komponens meghibásodási rátája λ.)

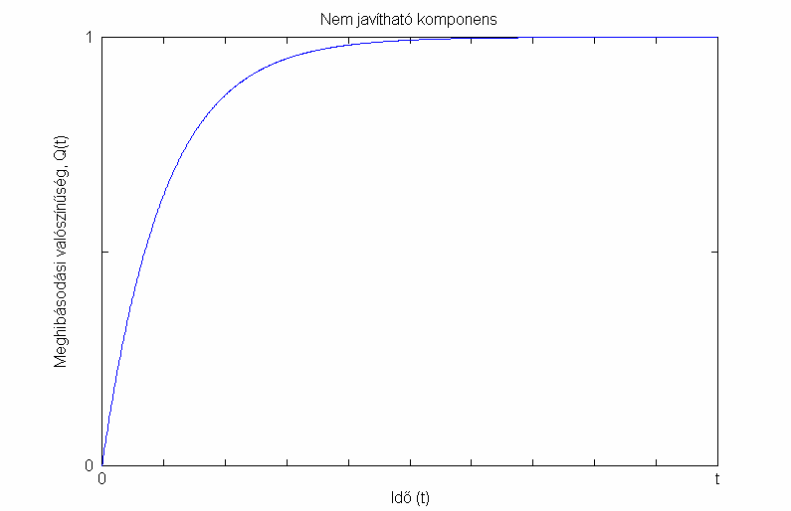
$$Q(t) = q + (1 - q) \cdot \left(1 - e^{-\lambda \cdot t}\right) \qquad \textbf{(5-9.)}$$

A képletben *q* a bekapcsoláskori meghibásodás valószínűségét jelenti.

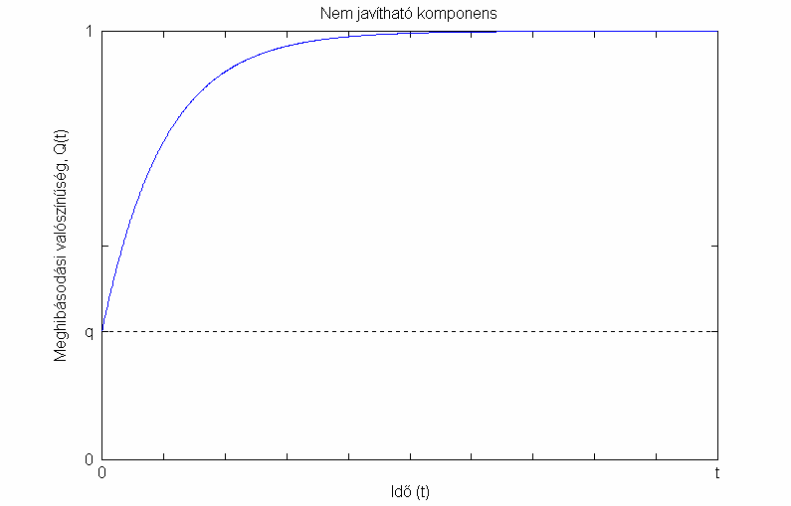
A rendelkezésre nem állás hosszú időre vett átlagát ennél a típusnál nincs értelme definiálni (és azzal számításokat végezni), mert az 1 lenne. Az időfüggést mutatják az 5-6. és 5-7. ábrák).



5-5 ábra: Konstans meghibásodási valószínűségű komponens



5-6 ábra: Nem javítható komponens Q(t) függvénye kezdeti meghibásodási valószínűség nélkül



5-7. ábra: Nem javítható komponens Q(t) függvénye kezdeti meghibásodási valószínűséggel

5.3.4 Folyamatosan ellenőrzött, javítható komponens

Az alkatrész bármilyen meghibásodása azonnal felfedésre kerül, és megindul a javítás. (A javítás jelentheti a komponens azonos típusú, hibátlan elemre történő lecserélését is.) A definiált javítási idő elteltével a komponens ismét hibátlanul működik, egészen a következő meghibásodásig.

A rendelkezésre nem állás időfüggvénye:

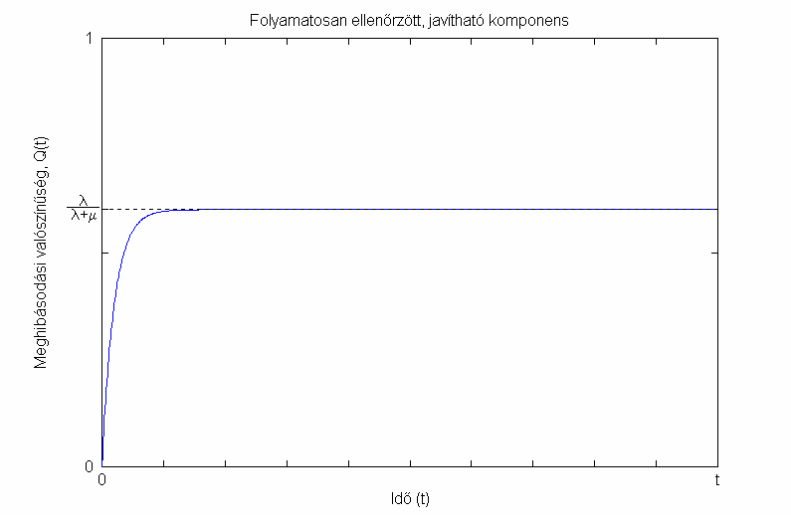
$$Q(t) = \left(\frac{\lambda}{\lambda + \mu} \right) \cdot \left(1 - e^{-(\lambda+\mu)\cdot t} \right) \qquad \textbf{(5-10.)}$$

ahol λ a komponens meghibásodási rátája és μ a karbantartási gyakoriság.

A rendelkezésre nem állás átlagos értéke:

$$Q_{átlag} = \frac{\lambda}{\lambda + \mu} \qquad \textbf{(5-11.)}$$

Az igen hosszú időre számolt átlagérték megegyezik az állandósult állapot értékével. Az időfüggést az 5-8. ábra mutatja.



5-8. ábra: Folyamatosan ellenőrzött, javítható komponens Q(t) függvénye

Amennyiben a bekapcsoláskori meghibásodási valószínűséggel is számolunk, az alábbi összefüggést kapjuk a rendelkezésre nem állásra:

$$Q(t) = q \cdot e^{-\mu \cdot t} + \left(\frac{\lambda}{\lambda + \mu} \right) \cdot \left(1 - e^{-(\lambda+\mu) \cdot t} \right) \qquad \textbf{(5-12.)}$$

Az összefüggés első tagja, $q \cdot e^{-\mu t}$ jelenti a bekapcsolási meghibásodásból származó részt. Látható, hogy a *q* bekapcsolási meghibásodási valószínűség hatása most nem marad meg tetszőleges ideig, mint az előző típusoknál, hanem a javítási időtől függően, exponenciálisan tart nullához, hiszen ennél a típusnál a bekapcsolási hibát is azonnal detektáljuk.

A rendelkezésre nem állás átlagos értéke ilyenkor is:

$$Q_{átlag} = \frac{\lambda}{\lambda + \mu} \qquad \textbf{(5-13.)}$$

5.3.5 Az időfüggvény módosítása megfigyelési adatok alapján

Az előző alfejezetekben bemutatott modellek egy-egy komponens rendelkezésre állását definiálták egy adott pillanatban (illetve az idő függvényében). Felmerül a kérdés: hogyan egyeztethetőek össze ezek a modellek a komponens megfigyelt állapotával? Hogyan oldható fel az a látszólagos ellentmondás, hogy a

komponensnek Q valószínűséggel hibásnak kellene lennie, de mi azt tapasztaljuk, hogy az működőképes (Q=0)?

A következő megállapítások csak állandó meghibásodási ráta mellett igazak, tehát nem érvényesek a beégetési és az öregedési szakaszra.

Vizsgáljuk meg a nem javítható komponens működőképességét:

$$R(t) = e^{-\lambda \cdot t} \qquad \textbf{(5-14.)}$$

Ugyanennek a komponensnek a működőképessége később:

$$R(t+\Delta t) = e^{-\lambda \cdot (t+\Delta t)} = e^{-\lambda \cdot t} \cdot e^{-\lambda \cdot \Delta t} = R(t) \cdot R(\Delta t) \qquad \textbf{(5-15.)}$$

Amennyiben valamilyen módon meggyőződünk arról, hogy a komponens a t időpillanatban működőképes (pl. teszttel, amelyet meghibásodás esetén javítás is követ):

$$R(t+\Delta t) = R(\Delta t) \quad ha \;\; R(t) = 1 \qquad \textbf{(5-16.)}$$

Ezt az összefüggést a periodikusan tesztelt komponens modelljének létrehozásához használhatjuk fel.

5.3.6 Periodikusan tesztelt komponens

Periodikusan tesztelt alkatrész hibái nem a hiba bekövetkezésekor, hanem egy előre definiált időpontban (előre definiált időintervallumonként periodikusan) lefutó tesztelés alkalmával fedődnek fel. A hiba felfedésekor azonnal megkezdődik a javítás, mely után az alkatrész ismét üzemképes lesz.

A periodikusan tesztelt komponens meghibásodási modellje talán a legkomplexebb modell, ezért több, a megadott paraméterek számának megfelelően különböző alesetekre bontottuk a vizsgálatát.

5.3.6.1 Csak a minimálisan megkövetelt paraméterek adottak (λ, TI)

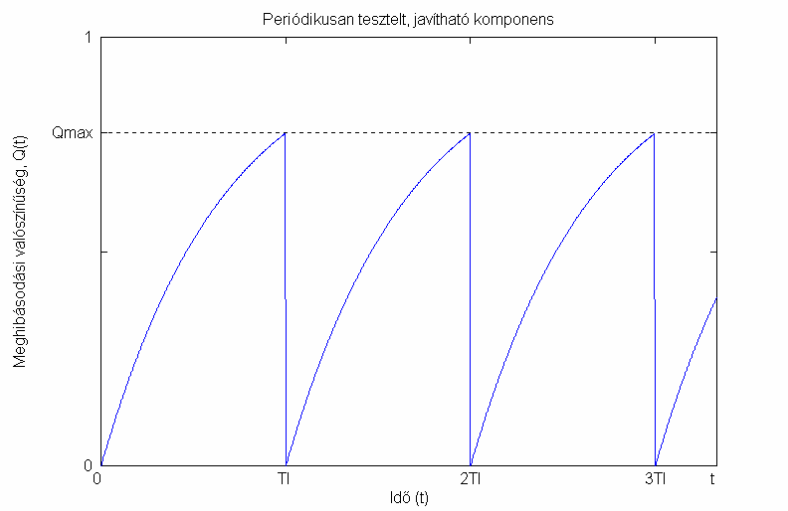
A legegyszerűbb esetben összesen két paraméter megadása szükséges: a komponens meghibásodási rátája (λ) és a tesztelési időintervallum (TI). Ilyenkor a javítási idő közel nulla hosszúságú (elhanyagolhatóan rövid), ami azt eredményezi, hogy a tesztelés után közvetlenül az alkatrész ismét hibátlan lesz. Ez a modell extrém rövid javítási idők esetén jól használható (5-9. ábra).

A rendelkezésre nem állás időfüggvénye:

$$Q(t) = 1 - e^{-\lambda \cdot (t - n \cdot TI)} \qquad ha \;\; n \cdot TI < t < (n+1) \cdot TI \qquad \textbf{(5-17.)}$$

Mivel Q(t) TI szerint periodikus, az átlagértéke:

$$Q_{átlag} = \frac{1}{TI} \int_0^{TI} Q(t)dt = 1 - \frac{1}{\lambda \cdot TI}\left(1 - e^{-\lambda TI}\right) \qquad \textbf{(5-18.)}$$



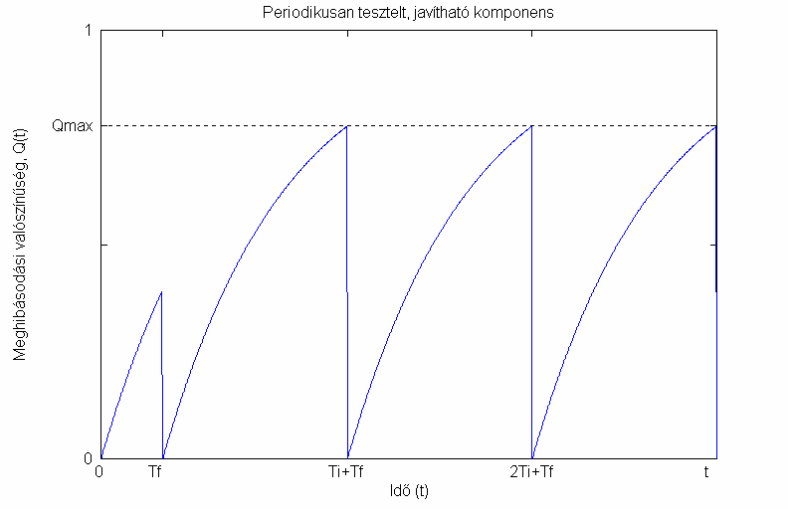
5-9. ábra: Periodikusan tesztelt komponens Q(t) függvénye

5.3.6.2 Az első teszt lefutásának időpontja különbözik a további tesztciklusoktól

Periodikusan tesztelt elemeknél gyakran alkalmazott az az eljárás, amikor a bekapcsolás utáni első teszt nem a megadott teszt-ciklusidő elteltével hajtódik végre, hanem annál jóval rövidebb idő után. Ennek kettős oka lehet: a bekapcsolás sok elemnél olyan igénybevételt jelent, amely megnöveli a meghibásodás valószínűségét, és ennek (illetve a bekapcsolás utáni működőképességnek) a detektálása szükséges. A másik ok a redundáns rendszerekben alkalmazott karbantartás- és megbízhatóság optimalizálásban keresendő, lásd a fejezet elején található bevezető példát. (A képletekben TF az első tesztig eltelő időszakot jelenti):

$$\begin{aligned} Q(t) &= 1 - e^{-\lambda \cdot t} && ha\ t < TF\ \ (T_i = 0) \\ Q(t) &= Q(TF) = 1 - e^{-\lambda \cdot TF} && ha\ t = TF \\ Q(t) &= Q(TI) = 1 - e^{-\lambda \cdot TI} && ha\ t = TF + n \cdot TI \\ Q(t) &= 1 - e^{-\lambda \cdot (t - n \cdot TI - TF)} && ha\ \ TF + n \cdot TI < t < TF + (n+1) \cdot TI \end{aligned} \qquad \textbf{(5-19.)}$$

Az időfüggést az 5-10. ábra mutatja.



5-10. ábra: Periodikusan tesztelt komponens Q(t) függvénye, ha TF<>TI

5.3.6.3 A javítási idő nem elhanyagolhatóan rövid, és az első teszt lefutásának időpontja különbözik a további tesztciklusoktól

Amennyiben eltérő első tesztidőközzel is számolunk, és a javítási időt már nem lehet elhanyagolni, az alábbi képletekhez jutunk (a képletekben TR a javítási időt jelenti):

$$
\begin{aligned}
&Q(t) = 1 - e^{-\lambda \cdot t} && ha\ \ t < TF \\
&Q(t) = Q(TF) = 1 - e^{-\lambda \cdot TI} && ha\ \ t = TF \\
&Q(t) = Q(TI) = 1 - e^{-\lambda \cdot TI} && ha\ \ t = TF + n \cdot TI \\
&Q(t) = Q(TI) + (1 - Q(TI)) \cdot \left(1 - e^{\lambda \cdot (t - n \cdot TI - TF)}\right) && ha\ \ TF + n \cdot TI < t < TF + n \cdot TI + TR \\
&Q(t) = 1 - e^{-\lambda \cdot (t - n \cdot TI - TF)} && ha\ \ TF + n \cdot TI + TR < t < TF + (n+1) \cdot TI
\end{aligned}
\qquad \textbf{(5-20.)}
$$

A rendelkezésre nem állás átlagos értéke:

$$
Q_{átlag} = 1 - \frac{1}{\lambda \cdot TI}\left(1 - e^{-\lambda \cdot TI}\right) + \left(1 - e^{-\lambda \cdot TI}\right) \times \frac{TR}{TI} \qquad \textbf{(5-21.)}
$$

Összevetve az 5-18 és az 5-21 képleteket, megállapíthatjuk, hogy nem végtelenül rövid javítás esetén a rendelkezésre állás valószínűsége csökken ($Q_{átlag}$ –ban megjelent egy additív tag.)

5.3.6.4 Az alkatrész bekapcsoláskor konstans meghibásodási valószínűséggel rendelkezik

Amennyiben a komponens konstans bekapcsolási (tesztelés utáni újbóli üzembe állítási) meghibásodási valószínűséggel rendelkezik, az alábbi képletek adódnak a rendelkezésre nem állásra:

$$
\begin{aligned}
&Q(t) = q + 1 - e^{-\lambda \cdot t} && ha\, t < TF \\
&Q(t) = Q(TI) = q + 1 - e^{-\lambda \cdot TI} && ha\, t = TF + nTI \\
&Q(t) = Q(TI) + (1 - Q(TI)) \cdot \left(q + 1 - e^{\lambda \cdot (t - TI)}\right) && ha\, TI < t < TI + TR \\
&Q(t) = q + 1 - e^{-\lambda \cdot (t - TI)} && ha\, TI + TR < t < 2TI
\end{aligned}
\qquad \textbf{(5-22.)}
$$

A rendelkezésre nem állás átlagos értéke:

$$
Q_{mean} = q + 1 - \frac{1}{\lambda \cdot TI}\left(1 - e^{-\lambda \cdot TI}\right) + \left(q + 1 - e^{-\lambda \cdot TI}\right) \times \frac{TR}{TI} \qquad \textbf{(5-23.)}
$$

5.3.6.5 A tesztelés alatt a komponens nem használható

Nagy megbízhatóságú, redundáns rendszereknél a redundancia fokától függően megengedhető, és a technológia sokszor szükségessé is teszi az olyan tesztelést, amely alatt a komponens (rendszer, alrendszer) a funkció-végrehajtás szempontjából nem elérhető, pl. mert valós jelek helyett teszt-jelekkel táplálják. Ilyen esetekben a

teszt időtartama alatt a rendelkezésre állás nulla értékű, függetlenül attól, hogy mekkora valószínűséggel következett korábban be meghibásodás.

$$\begin{aligned} Q(t) &= q + 1 - e^{-\lambda \cdot t} && ha\ \ t < TF \\ Q(t) &= Q(TI) = q + 1 - e^{-\lambda \cdot TI} && ha\ \ t = TF + nTI \\ Q(t) &= 1 && ha\ \ TI < t < TI + TR \\ Q(t) &= q + 1 - e^{-\lambda \cdot (t - TI)} && ha\ \ TI + TR < t < 2TI \end{aligned} \qquad \textbf{(5-24.)}$$

5.4 Komplex modellek

5.4.1 Bevezetés

Amennyiben egy alapeseménynél a 4. fejezetben leírtak szerint detektált és nem detektált meghibásodási valószínűségekkel dolgozunk, és szükségessé válik a rendelkezésre állás időfüggésének számítása is, az előző részben megismert időfüggést leíró modelleket adaptálni kell a problémakörhöz. Az ugyanazon alapeseményhez tartozó detektált és nem detektált meghibásodások nem foghatók föl két független eseményként, és így közvetlenül nem alkalmazhatók rájuk az előző fejezetben megismert modellek. Ugyanakkor a létrehozandó komplex modellek is az előző részben bemutatott klasszikus modellekből származnak [Gáspár és Szabó, 1998b].

A következő modelleknél az alábbi feltételezésekkel élünk:

- A teszt ciklusideje a rendszer meghibásodási rátájának, illetve egyéb jellemző időtartamainak ismeretében elhanyagolhatóan rövid.
- A teszt nem képes az összes lehetséges meghibásodást detektálni, hanem csak azok bizonyos százalékát (k).

5.4.2 Folyamatosan figyelt, nem javítható komponens modell

A modell olyan komponenseket ír le, amelyeknél javítást nem alkalmaznak (pl. megközelíthetetlen helyre beépített elemek, nukleáris érzékelők stb.), de a hibák bizonyos százaléka a fellépés után automatikusan detektálásra kerül.

$$P_{\det} = k\left(1 - \mathrm{e}^{-\lambda t}\right) \qquad \textbf{(5-25.)}$$

$$P_{un\,\det} = (1-k)\left(1 - \mathrm{e}^{-\lambda t}\right) \qquad \textbf{(5-26.)}$$

5.4.3 Periodikusan tesztelt, nem javítható komponens modell

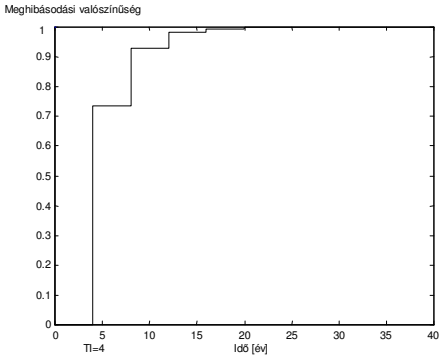
A modell a 5.4.2 pontban bemutatott modelltől abban tér el, hogy a meghibásodás felfedésére csak a periodikusan lefolytatott tesztek kínálnak lehetőséget.

$$P_{\det} = k\left(1 - \mathrm{e}^{-\lambda \cdot n \cdot TI}\right) \qquad \textbf{(5-27.)}$$

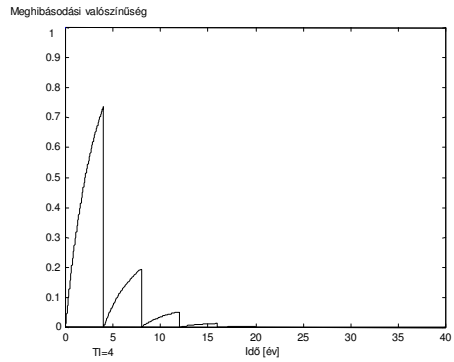
$$P_{un\,\det} = (1-k)\cdot\left(1 - e^{-\lambda\cdot(t - n\cdot TI)}\right) \qquad \textbf{(5-28.)}$$

ahol $n \cdot TI < t \le (n+1)\cdot TI$, TI a tesztelési periódus. A detektált és nem detektált hiba valószínűségét az 5-11. és az 5-12. ábrák mutatják.

Detektált hiba valószínűsége, ha k =1

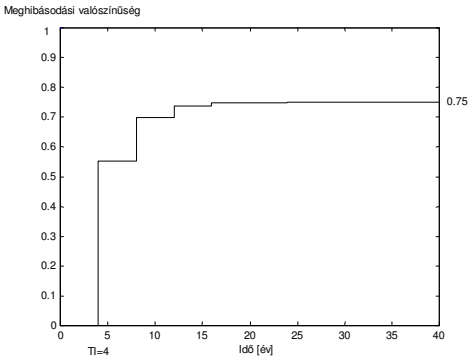


Nem detektált hiba valószínűsége, ha k =1

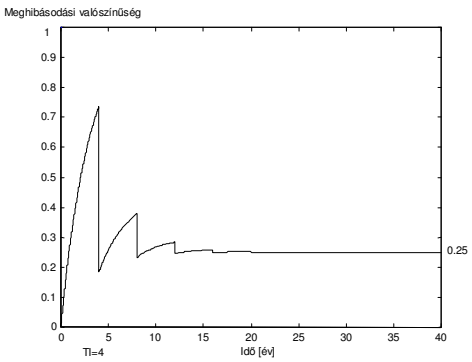


5-11. ábra: Periodikusan tesztelt, nem javítható komponens (k=1)

Detektált hiba valószínűsége, ha k =0.75



Nem detektált hiba valószínűsége, ha k =0.75



5-12. ábra: Periodikusan tesztelt, nem javítható komponens (k=0.75)

5.4.4 Periodikusan tesztelt, javítható komponens modell

A komponens periodikusan tesztelt. A detektált hibák javítására előre definiált időpontokban kerülhet sor, amelyek a teszt időpontoktól függetlenek is lehetnek (bár ez a megoldás műszakilag célszerűtlen). A javítási idő elhanyagolhatóan rövid.

A modell képletei az 5-27. és 5-28. képletekhez hasonlóak, de *t* helyett t_0 -t alkalmazunk az 5-29. képlet szerint:

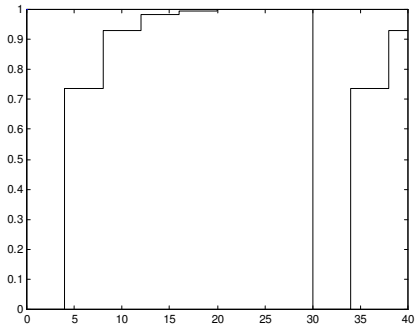
$$t_0 = t - m \cdot TRI \qquad \textbf{(5-29.)}$$

ahol $0 \le t_0 < TRI$, TRI a javítási intervallum (két javítás közötti időtartam), *m* pozitív egész. A nem detektált részt az alábbi összefüggés írja le:

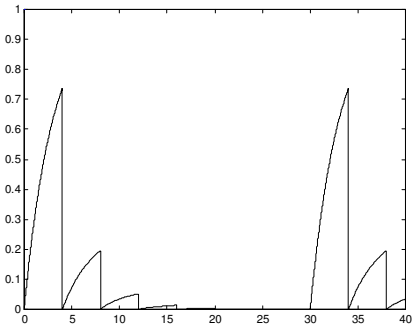
$$P_{un\,\det} = 1 - e^{-\lambda t_0} - k\ \left(1 - e^{-\lambda(m-1)T_i}\right) \qquad \textbf{(5-30.)}$$

A detektált és nem detektált hibavalószínűséget az 5-13. és az 5-14. ábrák mutatják.

Detektált hibavalószínűség, ha k =1

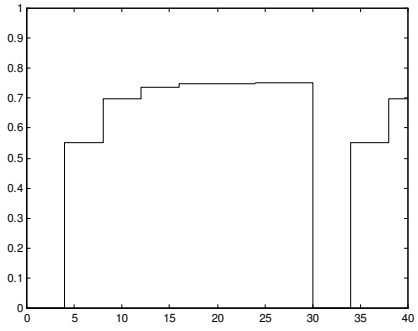


Nem detektált hibavalószínűség, ha k =1

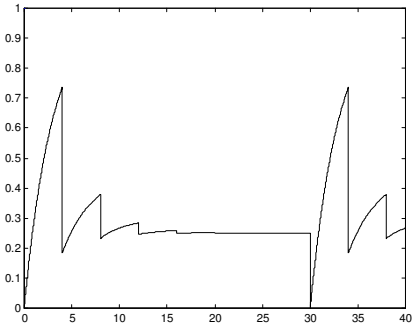


5-13. ábra: Periodikusan tesztelt, javítható komponens (k=1)

Detektált hibavalószínűség, ha k =0.75



Nem detektált hibavalószínűség, ha k =0.75



5-14. ábra: Periodikusan tesztelt, javítható komponens (k=0.75)

II. Tézis: Megállapítottam, hogy a hiba-aktív és hiba-adaptív logikákat tartalmazó rendszerek időfüggő megbízhatóság-elemzése komplex meghibásodási modelleket igényel.

II.A: Megállapítottam, hogy a komplex meghibásodási modelleknek a detektált és nem detektált meghibásodások valószínűségének időbeli változását, mint két összefüggő eseményt kell leírniuk.

II.B: Komplex meghibásodási modelleket hoztam létre a periodikusan tesztelt komponensek számára.

A komplex időfüggő modellek alkalmazása történhet akár az előző fejezetben bemutatott zárt alakú képletek segítségével is. Az időfüggés vizsgálata tradicionálisan az egyes időpillanatokban az időfüggő modell alapján meghatározott rendszerkomponens-megbízhatósági értékek rendszer-hibamodellbe való behelyezésével és a hibamodell kiértékelésével történik.

6. AUTOMATIKUS MODELLGENERÁLÁS

6.1 Bevezetés

A nagy megbízhatóságú rendszerek (atomerőművi védelmi rendszerek, vasútbiztosító berendezések, repülőgép fedélzeti rendszerek stb.) tervezésénél nagyon fontos tervezési-ellenőrzési lépés a megbízhatósági szint determinisztikus és/vagy valószínűségi alapú igazolása [Bokor et. al., 1997]. Ehhez az igazoláshoz sokféle módszertan került kifejlesztésre, pl. a Hibamódok és hatások analízise (FMEA), Markov analízis, Eseményfa analízis (ETA), Hibafa-analízis (FTA) stb. A módszerek közül egyértelműen a hibafa-analízist használják a legszélesebb körben, részben jól kidolgozott módszertana (lásd 3.8 alfejezet), részben a rendelkezésre álló szoftver eszközök miatt.

A hagyományos hibafa-analízis manuális hibamodell létrehozásán alapul. Ez az analízis-fázis mély rendszerismeretet, nagy rendszer- és analízis módszertani tapasztalatot igényel. Mindezek mellett a manuális modell felépítés időigényes, és így drága, valamint magában hordozza az emberi hibák, tévesztések lehetőségét. Az egész analízis-eljárás jelentősen gyorsítható, és hibamentessé tehető, amennyiben a modell létrehozása automatikusan történik. Különösen fontos lehet, hogy a hibamentes modellgenerálás verifikálható is, így az egyes analízisek hitelességének a bizonyítására a későbbieken már csak a generálás feltételeinek betartását kell ellenőrizni, illetve maga a teljes analízis is bármikor reprodukálható.

Az automatikus hibafa-generálás a rendszerváltozatok egységes kezelését is biztosítja, és így kimerítő analízist tesz lehetővé elfogadhatóan rövid idő alatt. Ennek egyik következményeként a rendszertervezés fázisában a megbízhatóság analízis kvázi on-line módon támogathatja a fejlesztési munkát, pl. fejlesztési változatok azonnali megbízhatóság-elemzésével.

Esetünkben az automatikus hibafa-generálást még egy problémakör motiválta: a korszerű, számítógép alapú biztonsági rendszerekben adaptív viselkedésű funkciókat alkalmaznak, amelyek modellezése a hibafa-analízisben csak igen terjedelmes rész-hibafákkal lehetséges, jelentősen lassítva és bonyolítva ezzel a modell-létrehozás fázisát, és ezen keresztül az analízist.

6.2 Kapcsolódó munkák

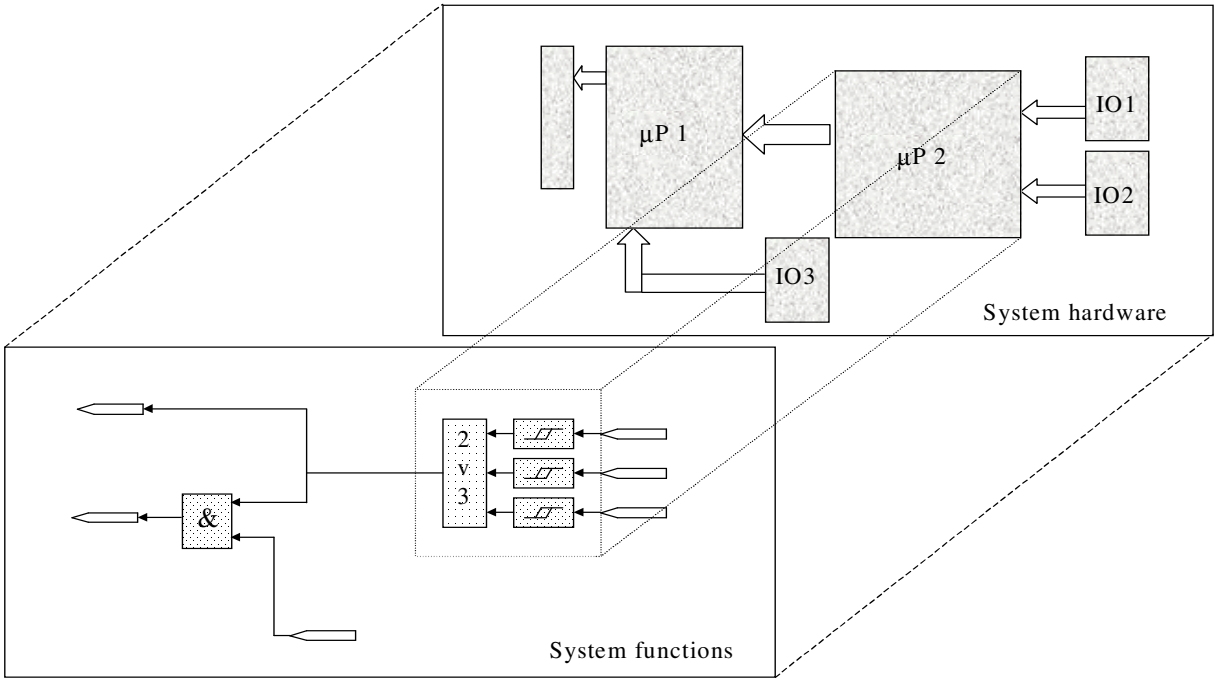
Természetesen a világon sok helyen foglalkoztatja a kutatókat és gyakorlati szakembereket az automatikus modell-létrehozás problémaköre. Napjaink legújabb eredményei között a kifejezetten számítógépek, illetve számítógépes rendszerek alacsony szintű (komponens szintű) modellezését lehetővé tévő (e mellett a tervezést is segítő) RIDL grafikus nyelv és a hozzá tartozó hibamodell generálás kifejlesztése [Verumi et. al., 1999], az erőművek mechanikus részeinek modellezéséhez kifejlesztett KB3 tudásalapú rendszer [Renault et. al., 1999], az analizálandó rendszer és környezete kapcsolatát modellező Formal Risk Analysis (FRA) módszer [Liggesmeyer és Rothfelder, 1998], valamint a rendszer-blokkdiagramm alapján dolgozó IRAS [Kocza és Bossche, 1997] említhető. Korábban keletkezett munkák közül megemlítjük [Lapp és Powers, 1977] algoritmusát.

A fenti munkák nagy mélységű, kimerítő analízist tesznek lehetővé, de nem veszik figyelembe a számítógépes alapú vezérlőrendszerek azon sajátosságát, hogy a megvalósított, magas szintű funkciók a hardver konfiguráció változtatása nélkül megváltoztathatók, valamint azt az ilyen rendszereknél felmerülő igényt, hogy az analízisnél az egyes rendszerkomponensek (processzor egységek, kommunikációs modulok stb.) már csak néhány meghibásodási paraméterrel legyenek modellezhetőek. Ezek az okok, valamint az előző alfejezetben bemutatott általános motiváció vezetett az alább bemutatandó eljárás kifejlesztéséhez [Gáspár és Szabó, 1999a], [Gáspár és Szabó, 1999b], valamint vasúti rendszerekben történő alkalmazásának vizsgálatához [Szabó és Tarnai, 2000].

6.3 Hardver és funkcionális rendszerleírás

Az elsődleges rendszerinformáció, amely a hibafa-analízis kiindulásaként is szolgál, a rendszer hardver topológiája. A topológia a rendszer felépítéséhez használt hardver elemeket tartalmazza, valamint leírja a közöttük lévő fizikai kapcsolatokat is. A korábbi években a topológia a rendszer funkcionalitását is meghatározta az elemek jól definiált, nem változtatható funkciója miatt.

Napjainkban, a számítógépes vezérlőrendszerek korában a rendszer topológiája és funkcionalitása különvált, és a végrehajtott funkciók a rendszer topológiájának változtatása nélkül is megváltoztathatók. Így szükség van egy külön leírásra, amely az aktuális funkcionalitást mutatja be, valamint a hardver elemek között létrejövő logikai kapcsolatokat. Egy tipikus példa az előzőekre egy olyan hálózati alkalmazás, amelyben minden egyes jelfeldolgozó egység a hálózatra csatlakozik, és a fizikai kapcsolatok segítségével bármely két feldolgozó egység között logikai kapcsolat hozható létre (de ezek nem szükségszerűen léteznek).



6-1. ábra: Funkcionális és hardver leírás

A hardver és a funkcionális rendszerleírás között szoros kapcsolat van, mivel minden egyes funkcionális elemet egy-egy jól definiált hardver egység valósít meg,

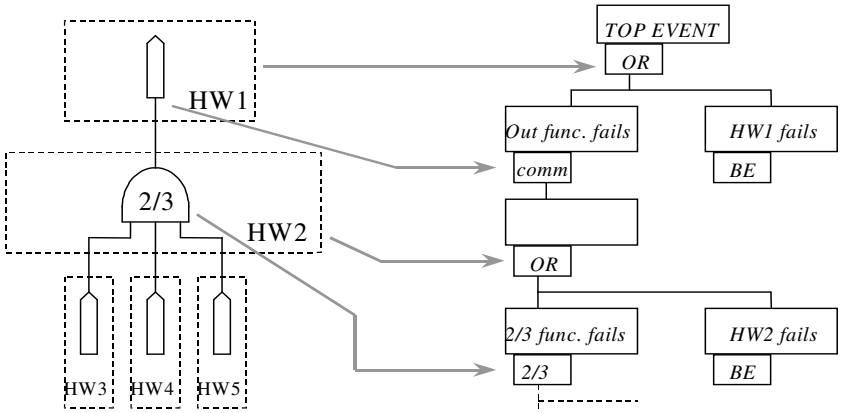
következésképpen minden funkcionális elemhez hozzá kell rendelni egy olyan paramétert, amely megadja a funkciót végrehajtó hardvert. A hardver és a funkcionális leírás úgy fogható fel, mint a teljes rendszerleírás két, összefüggő rétege (6-1. ábra).

A két rendszerleíró állomány elkészíthető csak a modell-létrehozás számára, szeparáltan, vagy generálható a rendszer fejlesztéséhez használt mérnöki tervező rendszer (pl. Siemens SPACE) bővebb információkat tartalmazó adatbázisa alapján.

6.4 Az automatikus modell-generálás algoritmusa

Az automatikus modell-generálás feladata a modellezendő rendszer meghibásodási szempontból vett viselkedését leíró hibafa létrehozása. A generálás a rendszer hardver és funkcionális leírásából indul ki, de ezek mellett egy meghibásodási viselkedések leírására szolgáló adatbázis, az un. szabálygyűjtemény használatára is szükség van. A szabálygyűjtemény módosításával a generált hibamodell nagysága, vagyis a hibaviselkedés leírásának mélysége változtatható. Részletes szabálygyűjtemény használata esetén a hibafa is részletes lesz, míg áttekintő elemzésekhez, ahol nincs szükség részletes modellekre, a nagyvonalú leírásokat tartalmazó szabálygyűjtemény is megfelelő.

Az automatikus hibafa-generálás az alábbi algoritmuson alapul (segítségül lásd a 6-2. és a 6-3. ábrát):



6-2. ábra: Hibafa-generálás

1. lépés: A rendszer azon pontjának kiválasztása, amelyre az analízist szeretnénk végrehajtani (a 6-2. ábrán kimeneti funkció). A modell-generálás típusának kiválasztása. Két fő analízis típust lehet megkülönböztetni, amelyek eltérő hibamodellt igényelnek: analízis a működés elmaradás vizsgálatára, valamint analízis a téves működések vizsgálatára. A két analízis típus eltérő szabálygyűjteményt igényel, mivel a hardver elemeknél különböznek a működést gátló jellegű meghibásodási módok és a téves beavatkozást okozó meghibásodási módok, valamint a funkcionális viselkedésben lévő logikai kapcsolatokat is másképpen kell modellezni a két esetben.

A csúcsesemény, a kiválasztott funkció nem megfelelő működése akkor következik be, ha az a hardver egység, amelyik a funkció végrehajtásáért felelős, nem működik megfelelően, ***vagy*** ha ugyanezen hardver egység hibátlan működése mellett nem kap megfelelő parancs (bemeneti) jelkombinációt. Következésképpen a hibafában

ennek modellezésére elágazás szükséges: létre kell hozni egy, a hardver meghibásodási módjait leíró ágat (HW ág), valamint egy funkcionális ágat, amely azokat a hibákat tartalmazza majd, amelyek következtében a funkció nem kap megfelelő bemeneti jelkombinációt (funkcionális ág). A két ág között OR hibafa-kapu teremt kapcsolatot.

2. lépés: A két ág létrehozása. A HW meghibásodásokat leíró ág a szabálygyűjteményben tárolt, az adott típusú hardver elem meghibásodási módjait és a szükséges paramétereket leíró adatok alapján tölthető fel. A másik ág (funkcionális ág) továbbágazik (vagy továbbágazhat) a modellezett logikai funkciónak megfelelően. A 6-2. ábrán az elsőként vizsgált funkció egy egyszerű un. kimeneti funkció, amely egy bemenettel rendelkezik. Így az a hibafa kapu, amely a funkcionális ágba kerül, és amelynek feladata az adott (jelen esetben kimeneti) funkció bemenő jeleiben bekövetkezett szabálytalanság és a funkció-végrehajtás sikeressége közötti kapcsolat modellezése, ebben az esetben egy egyszerű átkötés (vagy komment) kapu lesz (ha a bemenő jel nem megfelelő, a funkció kimenő jele sem lesz megfelelő).

3. lépés: Az aktuálisan modellezett funkció (esetünkben a kimeneti funkció) első bemenetének keresése (a példában ez egy 2/3 funkció). A továbbiakban az így kiválasztott funkciót kezeljük aktuálisként. Ha ennek a funkciónak a végrehajtásáért más hardver egység felelős, mint az előző funkció végrehajtásáért (esetünkben ez a helyzet), a hibafát két ágra kell szétválasztanunk ismételten, a hardver és a funkcionális ágra, a két ág között OR hibafa-kapu kapcsolattal (lásd az első lépésnél bemutatott okot). Ha az aktuális funkciót ugyanaz a hardver hajtja végre, mint az egy szinttel magasabban lévő funkciót, a hardver ág beszúrására nincs szükség (ez már a hibafa egy magasabb szintjén megtörtént).

A példában beszúrásra került a hardver ág. Ez után a funkcionális ágba az aktuálisan vizsgált (most 2/3) funkció hibaviselkedését leíró hibafa-kapu kerül. 2/3 funkció esetén a kimenet csak abban az esetben lesz nem megfelelő, ha kettő vagy több bemeneti jel nem megfelelő (hiszen két jó jel a háromból kettő funkció miatt elégséges a sikeres kimenethez), így a 2/3 kapu hibamodellezése 2/3 hibafa-kapuval történik.

4. lépés: Az aktuálisan végrehajtott funkció (2/3) első bemenetének megkeresése (a 6-2. ábrán egy bemeneti funkció). Ha az így megtalált funkció INPUT típusú (a példában az), további funkcionális modellezésre nincs szükség, csak az input funkciót megvalósító hardvernek megfelelő hardver ágat kell létrehozni, azt is csak akkor, ha a bemeneti funkciót nem ugyanaz a hardver hajtja végre, mint amelyik az előzőleg vizsgált funkciót. Ennek megfelelően a bemeneti funkciók a funkcionális leírás végeit jelentik.

Ha az aktuálisan megtalált funkció nem INPUT típusú lenne, ismételten vizsgálni kellene, hogy szükséges-e a hardver ág létrehozása, majd a funkcionális ágban modellezni kellene a hibaviselkedést is. Figyeljük meg, hogy innentől kezdve a 3. lépés ismétlődik egészen addig, amíg egy bemeneti funkcióhoz nem jutunk el.

5. lépés: Ha az aktuálisan vizsgált funkció bemenete INPUT funkcióhoz kapcsolódott, annak vizsgálata után, hogy szükség van-e a hardver ágra, egy speciális eljárás, az un. rollback funkció kerül végrehajtásra a generáló algoritmusban, mivel az INPUT a funkcionális leírás egyik végét jelenti, itt továbbmenni nem lehet. A rollback eljárás visszafelé megy a funkcionális leíráson, keresve egy olyan funkcionális elemet, amelynek még nem mindegyik bemenete volt feldolgozva/vizsgálva. Ennek a kivitelezéséhez minden egyes funkcionális leírásbeli elemhez egy számlálót rendelünk, amely azt mutatja meg, hogy a vizsgálat a funkcionális elem melyik bemenete irányában folytatódott. Ha ez a szám megegyezik az összes bemenet számával, az azt jelenti, hogy a funkció teljes egészében modellezésre került, és a rollback eljárás eggyel magasabb szintre térhet vissza. Ha a számláló értéke kevesebb az összes bemenet számánál, az érték eggyel növelődik, majd a 3. lépés kerül ismételten végrehajtásra, csak most nem a funkció első bemenetére, hanem a számláló által leírt sorszámúra. (Pontosan fogalmazva a 3. lépés mindig a funkció első, még nem modellezett bemenetére kerül végrehajtásra.) A hibafába új ág beszúrása szükséges, ha találtunk kaput még nem modellezett bemenettel. Hogy pontosan megmondható legyen, a hibafát melyik ponton kell folytatni, minden egyes funkcionális leírásbeli elemhez egy-egy mutatót kell rendelni, amelyik megmutatja, melyik hibafa-kapu modellezi az adott funkció bemeneti meghibásodásai és kimeneti meghibásodása közötti viszonyt.

A modellezési folyamat akkor fejeződik be, amikor a rollback eljárás vissza tud térni a kiindulási funkcióhoz úgy, hogy annak is minden bemenete feldolgozásra került.

6. lépés: Mivel az algoritmus egy nemstruktúrált hibafát szolgáltat, a megjelenítés és nyomtatás céljaira a hibafát lapokra kell tördelni. E célból un. transzfer eseményeket kell a hibafa megfelelő helyeire beszúrni.

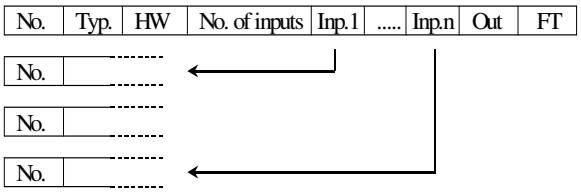
A 6-3. ábra az algoritmust mutatja magas szintű, Delphi (Pascal) alapú programnyelven leírva.

```
HW:= NONE                          // Starting conditions.
FUNC:= TOP^                        // Output specified for the analysis.
REPEAT                             //
  If HW(FUNC) <> HW                // If the function belongs to a new HW,
      {PUTFT('OR')                 //    place an OR gate to the FT
       PUTFT(FAULTS(HW(FUNC)))     // HW failure struct (HW branch).
       SEARCH_ROUTE(HW(FUNC),HW)   // Searching for elements between the two hw
      HW:=(HW(FUNC))          }    // The new HW stored as default.
  PUTFT(FTGATE(TYP(FUNC)))         // Place a gate to FT (functional branch).
  If TYP(FUNC)='INP'  { ROLLBACK } // Searching for the next path.
      ELSE   {GET_NEXTINP(FUNC) }  // Get an input of the gate.
UNTIL ROLLBACK_TO_TOP              // End of the procedure.
```

6-3. ábra: Az egyszerűsített algoritmus

A fentebb leírtaknak megfelelően a rendszerleíró adatbázis két táblát tartalmaz: a hardver leíró és a funkcionális leíró táblákat. Mindkét táblában a rekordok pointereket tartalmaznak annak érdekében, hogy a bemenetektől a kimenetek felé tartó láncolt listák jöjjenek létre (6-4. ábra). A funkcionális leírásban a blokk bemeneteit és kimenetét azonosító pointerek mellett egyéb pointerekre is szükség van. Az egyik azzal a hardver elemmel köti össze a funkciót, amelyik a funkciót hajtja végre, míg a

másik, amely a hibafa-generálás során kap értéket, a funkciót modellező hibafa-kapura mutat a hibafát leíró táblában.



6-4. ábra: A funkcionális leírás adatbázis formátuma

A hibamodell-generáláshoz a felhasználónak szabályokat kell létrehoznia, amelyek leírják a hardver elemek lehetséges meghibásodási módjait és a meghibásodási modellek alkalmazásához szükséges paraméterek értékeit, úgymint meghibásodási ráta, javítási idő, tesztelési idő stb. A szabálygyűjteményre mutat egyszerűsített példát a 6-5. ábra.

```
Object Processor_type_A ;
  Failure modes :
    ”Processor_type_A %1 fails, detected”,
    model_1, FR=2.5E-6, Repair=24h;
    ”Processor_type_A %1 fails, undetected”,
    model_2, FR=7.4E-7, Repair=24h, Test=1000h;
  If connected to:
  Subrack_type_A, add
    ”Processor_type_A %1 blocks back plane bus”,
    model_1, FR=1E-6, Repair=24h;
  Ethernet_type_10BaseT, add
    ”Processor_type_A %1 blocks network communication”,
    model_1, FR=1E-6, Repair=24h;
End of object Processor_type_A;

Object Input_binary ;
  Failure modes :
    ”Binary input module %1 one channel fails, detected”,
    model_1, FR=3.4E-7, Repair=24h;
    ”Binary input module %1 one channel fails, undetected”,
    model_2, FR=2.4E-7, Repair=24h, Test=2500h;
    ”Binary input module %1 (all channel) fails, detected”,
    model_1, FR=5.1e-6, Repair=24h;
    ”Binary input module %1 (all channel) fails, undetected”,
    model_2, FR=2E-6, Repair=24h, Test=2500h;
  If connected to:
  Subrack_type_A, add
    ”Binary input module %1 blocks back plane bus”,
    model_1, FR=1E-6, Repair=24h;
End of object Input_binary;
```

6-5. ábra: Egyszerűsített szabálygyűjtemény

A fentebb bemutatott algoritmus elsődlegesen nukleáris erőművek védelmi rendszereinek megbízhatósági analíziséhez került kifejlesztésre. A védelmi rendszerek általában egyszerű számítási és beavatkozási utakat tartalmaznak. Ha a funkcionális leírás hurkokat is tartalmaz, az algoritmus módosítása szükséges a hurkok kezeléséhez. A hurkok detektálása kivitelezhető annak a számlálónak a használatával, amely az aktuálisan feldolgozott bemenet számát hivatott tárolni minden egyes funkcionális leírásbeli elemnél. Ezeket a számlálókat a rollback eljárás nullázza akkor, amikor az adott elemnél az összes bemenet feldolgozása megtörtént, és lehetséges az egy szinttel feljebb lépés. Ha a leíráson lefelé haladva, vagyis

akkor, amikor egy funkcionális elem bemenetét tápláló mások funkcionális elemet kezdünk vizsgálni (3. lépés az algoritmusban), ennek a számlálónak nullán kell állnia, ellenkező esetben hurkot találtunk. Az automatikus modell-generálás a hurkok tekintetében manuális beavatkozást igényel, pl. a hurok egy egyedi funkcionális elemmel való modellezését, amennyiben ez lehetséges.

6.5 Speciális esetek

Ahogy a korábbiakban bemutattuk, a funkcionális és a hardver leírások között szoros kapcsolat van. Minden funkcionális elemhez egy és csakis egy hardver egység tartozik, amely felelős a funkció végrehajtásáért. Ugyanakkor a hardver elemek több funkcionális egységet is tartalmazhatnak (a gyakorlatban nagyon sokat tartalmaznak), de lehetnek olyan hardver egységek is, amelyek funkciói rendszerszintűek, és a felhasználói funkcionális leírásban nem jelennek meg. Ilyen egységek pl. a kommunikációs kapcsolatot biztosító elemek.

A több funkció végrehajtásáért felelős hardver egységek az algoritmusban nem okoznak problémát, de azok az egységek, amelyek a jelútban fekszenek, és funkciójuk nem jelenik meg a funkcionális leírásban, külön kezelést igényelnek. Az alapalgoritmus csak azt vizsgálta, hogy két, egymással kapcsolatban lévő funkcionális elem azonos hardver egységben kerül-e végrehajtásra, és ha nem, akkor új hibafa-ágat hozott létre az új hardver egység meghibásodási módjainak kezeléséhez. A felhasználói funkció nélküli hardver elemek miatt vizsgálni kell azt is, hogy a két, különböző hardver egység vajon fizikai kapcsolatban van-e egymással. Ha nem, meg kell keresni a köztük kapcsolatot létesítő hardver elemeket. Ezt a keresést egy eljárás, a SearchRoute eljárás valósítja meg (lásd 6-3. ábra). A SearchRoute eljárás először a szabálygyűjtemény egy szekcióját olvassa végig. A végigolvasott szekció tartalmazhat direkt módon megadott kapcsolatokat. Amennyiben itt nem található olyan összerendelés, amely alapján a két kérdéses hardver elem közötti kapcsolat megállapítható, a hardver leírásban az egyik elemtől elindulva a legrövidebb (legkevesebb közbenső elemet tartalmazó) utat keresi meg. Amennyiben a két elem között vannak további hardver elemek, azok meghibásodási szempontból soros rendszert képeznek az információáramlás szempontjából korábban lévő elemmel, és a meghibásodásukat reprezentáló hibafa-kapukat ugyanazon helyre kell beszúrni a hibafába (VAGY kapcsolatban a korábbi hardver elem meghibásodásait leíró hibafa kapukkal vagy alapeseményekkel), ahova a korábbi hardver miatti hardver ág került.

A felhasználói funkció nélküli hardver egységeknek van egy másik csoportjuk is. Ezek az elemek más hardver egységek kiszolgálását végzik, mint pl. a tápegység-modulok. Ezen egységek meghibásodásainak is be kell kerülniük a hibafába, hiszen a rendszer működésére hatással vannak. Erre szolgál a következő módszer: a hardver egységek leírásánál megadható egy *if connected to …* szekció, amelyben leírható, hogy amennyiben az adott egységhez más típusú egység is csatlakozik (pl. tápegység), a csatlakozó hardver elem meghibásodását reprezentáló alapesemények is bekerüljenek a hibafába. A fenti eljárás használható olyan esetekben is, ha felhasználói funkciókkal rendelkező hardver egységek hathatnak egymásra fizikailag.

A vezérlőrendszereknél gyakori, hogy a funkcionális leírás olyan elemeket tartalmaz, amelyeknek egy bemenete és egy kimenete van, vagyis meghibásodási szempontból nem teremt kapcsolatot jelek és így távolabbi hibaokok között (pl. határértékképző

vagy komparátor). Ezeket az elemeket a modell-generálás során egyszerűen át lehet ugrani, amit a generálási szabálygyűjteményben adhatunk meg az algoritmus számára (a funkcionális elemnek megfelelő hibafa-elem típusa *NONE*). Ez a megoldás akkor is alkalmazható, ha a kihagyott elemet egyedi hardver futtatja, de alkalmazzuk az előző részben bemutatott SearchRoute útkereső algoritmust.

III. Tézis: Megállapítottam, hogy a számítógépes alapú ipari vezérlőrendszerek hibafa-modelljének generálása automatikus módon, a funkcionális specifikáció elsődleges feldolgozásával is lehetséges.

- **III.A: Megállapítottam, hogy az automatikus modell-létrehozás igényli a rendszer hardver és funkcionális leírását, és a két leírás közötti kapcsolatok megadását is. A leírásoknak formalizáltaknak kell lenniük a feldolgozhatóság érdekében.**
- **III.B: Megállapítottam, hogy a modell-generálás a funkcionális leírás ágainak bejárásával, az egyes funkciókat végrehajtó hardver egységek, illetve egységhatárok figyelésével és hibamodellezésével valósul meg. A hibamodellezés formalizált szabályalap segítségével történhet.**
- **III.C: Algoritmust hoztam létre az automatikus hibafa-generálás megvalósítására.**
- **III.D: Az algoritmus működőképességét RiskSpectrum környezetbe integrált programcsomag segítségével ellenőriztem.**

6.6 Megvalósítás

A fenti hibafa-generáló algoritmus tesztelésére 32 bites Windows operációs rendszerekhez Borland Delphi programnyelven programot fejlesztettünk, amely RiskSpectrum hibafa-analízis program számára generált strukturálatlan hibafákat.

A RiskSpectrum hibafa-analízis program [RiskSpectrum a],[RiskSpecrtum b] a svéd Relcon cég [Relcon] terméke, a világon széles körben ismert és elfogadott. Atomerőművi alkalmazások számára kvázi szabvány ennek a programnak a használata. További előnye az analízisprogramnak, hogy hozzá a fejlesztő cég ingyenesen letölthető, csak a hibafák megtekintésére alkalmas un. viewer programot biztosít.

A RiskSpectrum program MS Access adatbázist használ a hibafák adatainak tárolására, ezt az adatbázis nyitja meg és tölti fel megfelelő elemekkel a mi hibafa-generáló programunk a fentebb bemutatott algoritmusnak megfelelően. A generáláshoz szükséges hardver és funkcionális leírás szöveges alapú, az egyes objektumok típusát és kapcsolatait sorolja fel.

Az algoritmus megvalósításáról és a konkrét program használatáról további részletek találhatóak [Szabó és Csiszár, 2000a], valamint [Szabó és Csiszár, 2000b]-ben.

6.7 Alkalmazási eredmények

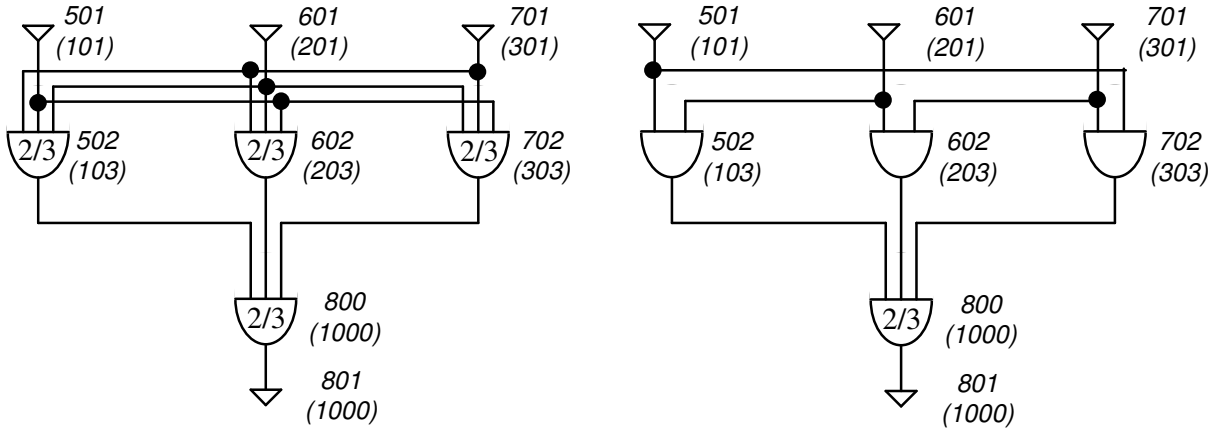
A következő példában bemutatjuk az algoritmus működését egy egyszerű példán. A példában egy feltételezett erőművi védelmi rendszer két funkcionális megvalósítását modellezzük. A példabeli rendszer háromszoros redundanciával épült (az egyes redundanciák elnevezése: train), a három train közel azonos felépítésű. Mindegyikben egy-egy szenzor ad jelet egy jelfeldolgozó/komparátor egységnek, amely a szomszédos trainekhez kapcsolódó érzékelők jelét is megkapja. A jelfeldolgozó egység a feldolgozás eredményét egy kimeneti egység segítségével

továbbítja a szavazó áramkörnek, amely beavatkozó jelet generál a technikai folyamat vezérléséhez.

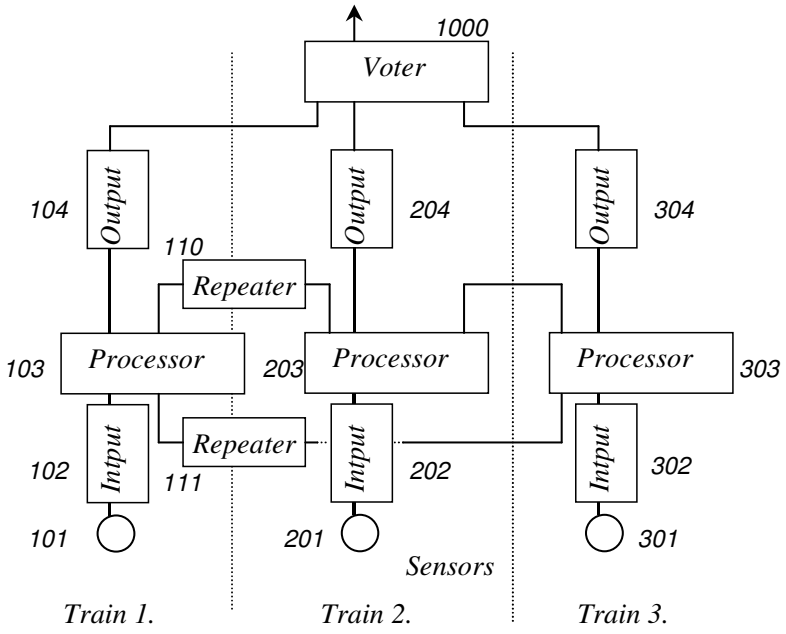
A példában a 2. és a 3. train közötti fizikai távolság kicsi, így a jeltovábbítás ismétlők nélkül történik, míg az 1. train az installálási távolságok miatt jelismétlő áramkörökön keresztül kommunikál a két szomszédos egységgel.

A példa második részében tovább egyszerüsítjük a rendszer funkcióit, és az egyes jelfeldolgozó egységek csak két szenzor jelét kapják meg a rendszerben lévő logikai kapcsolatok számának csökkentése érdekében.

A rendszer funkcionalitását a 6-6. ábra mutatja, a hardver kialakítás a 6-7. ábrán látható. A funkcionális rajzon az elemek azonosítóját, illetve zárójelben az adott funkcionális elemet megvalósító hardver egység azonosítóját is feltüntettük. A hardver egységek azonosítói természetesen a hardver ábrán is megtalálhatóak.



6-6. ábra: A demonstrációs példa rendszerének funkcionalitása



6-7. ábra: A demonstrációs példa rendszerének HW terve

A példabeli rendszer két különböző funkcionalitásához generált hibafák, valamint az alkalmazott szabálygyűjtemény és rendszerleírás a 2. mellékletben találhatóak.

7. NAGY MEGBÍZHATÓSÁGÚ RENDSZEREK KÖZLEKEDÉSI ALKALMAZÁSAI

7.1 Bevezetés

A különböző közlekedési ágazatok (vizi-, légi-, közúti és vasúti közlekedés) eltérő megbízhatósági igényeket támasztanak a járműveken és az irányításban alkalmazott rendszerekkel szemben. A biztonság szempontjából legkritikusabb két alkalmazás a repülőgépek fedélzeti berendezései és a vasúti irányítás berendezései (biztosítóberendezések).

A repülőgépek fedélzeti berendezéseinél nem lehet olyan rendszerállapotot kijelölni, amelynek elérése biztonságot eredményez, így ezeknek a berendezéseknek a működőképességét mindenképpen fenn kell tartani. Ez a cél hibatűrő redundáns rendszerekkel érhető el.

A vasúti biztosítóberendezési technikában ezzel szemben biztonsági állapotnak fogadják el azt az állapotot, amikor nincsen vonatmozgás. A biztosítóberendezésben fellépő meghibásodás esetén a rendszer leállítása és a biztonsági állapot felvétele megfelelő reakció. Ezt a viselkedésmódot a hibabiztos rendszerek valósítják meg. Ugyanakkor meg kell jegyezni, hogy a folyamatos üzem fenntartása ebben az esetben is fontos lehet: nem biztonsági szempontból, hanem rendelkezésre állási (gazdaságossági) szempontból.

A következő részekben a vasúti biztosítóberendezések biztonságosságát és rendelkezésre állását fogjuk vizsgálni.

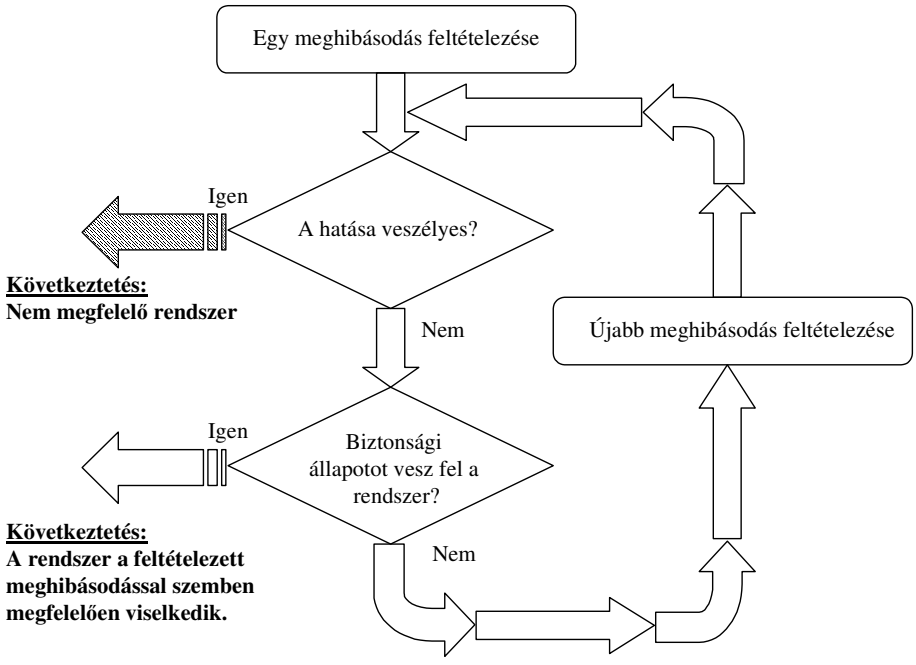
7.2 A vasúti biztosítóberendezések vizsgálati eljárásai

Új biztosítóberendezések létesítésénél, vagy módosított berendezések újbóli üzembe helyezésénél a berendezés specifikációnak való megfelelőségét bizonyítani kell. Az eljárás neve érvényesítés (validation). A megfelelőség bizonyítása kiterjed a specifikáció mindhárom területére: bizonyítani kell a funkcionális megfelelőséget, a műszaki követelmények teljesítését és a megbízhatóságra vonatkozó követelmények teljesülését is. A funkcionális és a műszaki megfelelőség bizonyítása a hardver egységek és a kapcsolódó berendezések megfelelő, hibamentes működését feltételezi. Ezzel szemben a megbízhatósági követelmények teljesítésének vizsgálata a berendezés részegységeinek, alkatrészeinek bekövetkező meghibásodásait tételezi fel, és ilyen feltételek mellett vizsgálja a rendszer működését [Hartonas-Garmhausen et. al.,1998], [Hudoklin és Rozman, 1985], [Jensen, 1996], [Tarnai a], [Tarnai b], [Tarnai c], [Görög et. al., 1998].

A következőkben bemutatjuk a Magyar Államvasutaknál (MÁV) alkalmazott megbízhatósági vizsgálati eljárásokat. Annak eldöntésére, hogy egy adott vizsgálat során milyen eljárást szükséges használni, részben törvényi (szabvány) előírások, részben pedig hatósági (vasúti területen a Közlekedési Főfelügyelet) rendeletek adnak útmutatást.

7.2.1 Determinisztikus vizsgálat

A biztosítóberendezési technikában alkalmazott determinisztikus vizsgálati eljárás nagyon hasonló a korábban bemutatott FMEA vizsgálathoz, de annál komplexebb. A vizsgálat első lépései megegyeznek a klasszikus FMEA első lépéseivel: a rendszer alrendszerekre bontása, amennyiben a komplexitás, illetve a vizsgálat tervezett mélysége azt indokolja; az egyes, a rendszerben alkalmazott alkatrésztípusok lehetséges meghibásodási módjainak összegyűjtése, definiálása (adott esetben ez a lépés általános műszaki előírásokon, hibakatalógusokon alapul). Ezt követően a rendszerbeli egyszeres meghibásodások elemzése következik annak eldöntésével, hogy a meghibásodás veszélyes hatású-e avagy nem, detektálásra kerül-e avagy sem. Az elemzés során a meghibásodás detektálása azért kap kiemelt szerepet, mert a feltételezés szerint ilyenkor automatikusan a biztonsági állapot felvétele történik meg (természetesen a legkorrektebb az, ha az analízis a detektáltság ténye helyett a biztonsági állapot felvételére összpontosít). Az igazi különbség az FMEA eljárás és a biztosítóberendezési technikában használt elemzési módszer között az eljárás folytatásában van: Az egyszeres meghibásodás elemzése további, járulékos meghibásodások feltételezésével kell, hogy folytatódjék, ha az egyszeres meghibásodás nem veszélyes hatású, de nem is hoz létre a további meghibásodások esetén is fennmaradó biztonsági állapotot. A vizsgálat azonban nem követeli meg a biztonsági állapot azonnali felvételét a meghibásodás bekövetkeztekor, megelégszik azzal, ha ez bekövetkezik akkor, amikor olyan funkció kerül végrehajtásra, ami igényli a meghibásodott hardver elem működését is (ez azt jelenti, hogy a meghibásodás a funkcióvégrehajtás szempontjából akadályozó jellegű). Az eljárás folyamatát a 7-1. ábra mutatja. Egy tetszőleges meghibásodás hatásának analízise csak akkor fejeződik be, ha a meghibásodás (és az esetlegesen mellé feltételezett további meghibásodások együttesen) biztonsági állapotot hoznak létre (ekkor a viselkedés megfelelő) vagy veszélyes állapotot okoznak (ekkor az egész rendszer vizsgálata sikertelen).



7-1. ábra: A determinisztikus vizsgálat folyamata

A determinisztikus eljárás előnye, hasonlóan az FMEA eljáráshoz, hogy bottom-up (alulról /a rendszer elemi szintjeitől/ felfelé végrehajtott) típusú analízis, így nem szükséges rendszerszinten meghibásodási eseményeket definiálni, és azok bekövetkezését elemezni, hanem az alapszintű (elem- vagy alrendszer) meghibásodások elemzése azokat automatikusan szolgáltatja.

Ezen előnyök mellett azonban vannak gyenge pontjai is a módszernek. A probléma elsősorban a biztonsági állapot elérésének, illetve az akadályozó jelleg kifejtésének időpontjában keresendő. Nem kerül vizsgálatra ugyanis az, hogy mennyi idő telik el a meghibásodás bekövetkezése és a biztonsági állapot felvétele, vagy a meghibásodás bekövetkezése és a következő működési igény fellépte (ekkor mutatkozik meg az akadályozó jelleg) között. Amennyiben a kérdéses idő nagy, ezen idő alatt is számolni kellene további esetleges hibák fellépésével.

7.2.2 Determinisztikus vizsgálat valószínűségi adatokkal

A determinisztikus vizsgálat valószínűségi adatokkal való támogatása (pl. MŰ8004-es német vasúti irányelven alapuló vizsgálati módszer [MÜ8004]) megoldást kínál a determinisztikus eljárás gyenge pontjára azzal, hogy bevezeti a hibafeltárási idő fogalmát. A hibafeltárási idő az az időtartam, amely alatt a fellépett meghibásodást detektálni (és a hatását semlegesíteni) kell, ellenkező esetben a tolerálhatónál nagyobbra növekszik egy újabb meghibásodás valószínűsége. A hibafeltárási idő elnevezés helyett a további részekben a második hiba fellépési idő (second error occurence time) fogalmát használjuk, mivel ez szemléletesebben mutatja azt a tényt, hogy ezen időtartam eltelte után szükséges csak újabb meghibásodással számolnunk. Hasonlóan az előzőekhez, második hiba fellépési idő alatt azt az időtartamot értjük, amely alatt, egy tetszőleges meghibásodás bekövetkezése után még kellően csekély egy újabb meghibásodás bekövetkezésének valószínűsége. Természetszerűleg ez az időtartam függ a meghibásodások bekövetkezésének gyakoriságától, így nem tekinthető tisztán determinisztikusnak, inkább a valószínűségi adatok figyelembe vétele miatt kevert eljárásnak (7-2.ábra).

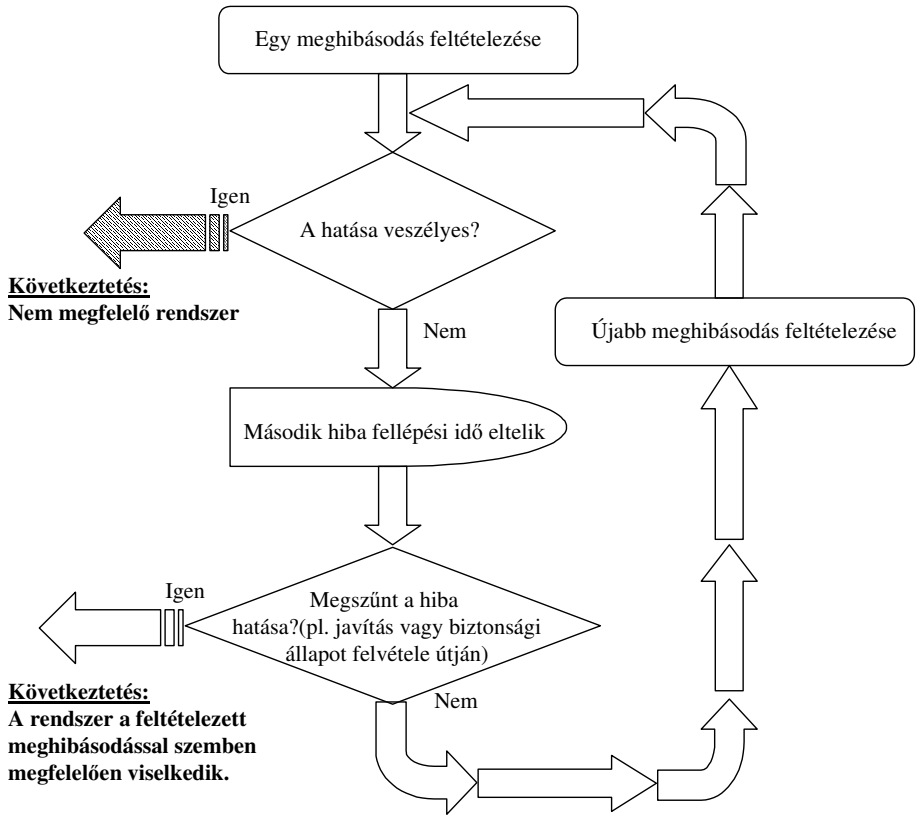
A MÜ8004 a második hiba fellépési idő számítását függővé teszi attól, hogy hány komponens meghibásodása lenne szükséges egyidejűleg a veszélyes állapot kialakuláshoz. Három esetet deklarál: két egyidejű meghibásodás esetén kialakulhat veszélyes állapot; csak három egyidejű meghibásodás hatásaként alakulhat ki veszélyes állapot és végül csak háromnál több egyidejű meghibásodás esetén alakul ki veszélyes állapot.

A MÜ8004 a második hiba fellépési idő számítására az alábbi összefüggést adja, amennyiben két egyidejű meghibásodás hatása veszélyes lehet:

$$T_{Second\ Error\ Occurrence} = \frac{1}{1000a} \qquad \textbf{(7-1.)}$$

ahol *(a)* azon komponensek vagy alrendszerek meghibásodási rátáinak összege, amelyek együttesen veszélyes állapotot hozhatnak létre. A fenti képlet abban az esetben alkalmazandó, amennyiben a funkció-végrehajtásban érintett komponensek közül bármely kettő meghibásodása veszélyes állapotot hozhat létre. A vizsgálat során most azt kell elemezni, vajon a meghibásodás (vagy meghibásodások) hatása biztonsági állapotot hoz-e létre a második hiba fellépési időn belül, vagy detektálódik és elhárításra kerül-e ugyanezen periódus alatt. Amennyiben a hibadetektálás

megtörtént, már csak azt kell biztosítani, hogy a detektálási és javítási periódus a második hiba fellépési időnél rövidebb legyen, amit átlagos megbízhatóságú alkatrészek és átlagos rendszerstruktúra esetén meg lehet valósítani.



7-2. ábra: A determinisztikus vizsgálat folyamata valószínűségi adatok használata esetén

7.2.3 Valószínűségi alapú vizsgálatok

A valószínűségi vizsgálatok célja (összhangban a korábban bemutatottakkal) egy (vagy több) nem kívánatos esemény bekövetkezési valószínűségének vagy gyakoriságának meghatározása. Alapmódszerként sok megbízhatósági analízis technika szóba jöhet, mint pl. az FTA, az ETA stb. Az új európai szabványok támogatják ezeket a módszereket [EN 50126], [EN 50129].

A vasúti vizsgálatoknál is az általános analízislépéseket kell követni:

1. Az egyes rendszerkomponens-típusok meghibásodási módjainak összegyűjtése.
2. Paraméterek összegyűjtése: meghibásodási ráták, tesztelési intervallumok, javítási intervallumok stb.
3. Hibaviselkedési modell létrehozása a rendszerre.
4. A modell analízise.

Az analízis eredménye egy valószínűségi vagy gyakorisági számérték, amelyet a korábban felállított követelményszinttel összehasonlítva megállapítható a rendszer megfelelősége vagy meg nem felelősége.

7.3 Valószínűségi határértékek képzése a biztosítóberendezések valószínűségi alapú vizsgálatához

7.3.1 Definíciók

A biztosítóberendezések valószínűségi alapú vizsgálatához definiáljuk ismét a rendelkezésre állás (availability), és a biztonság (safety) fogalmát, illetve ezek ellentétjeit, a rendelkezésre nem állás (unavailability) és a veszélyeztetettség (biztonság hiánya - unsafety) fogalmát.

Rendelkezésre állás (A): Annak valószínűsége, hogy a berendezés egy adott időpontban képes a specifikációjában rögzített feladatok és követelmények teljesítésére.

Biztonság (S): Annak a valószínűsége, hogy a berendezés egy adott időpontban nem okoz veszélyes állapotot sem a vezérelt folyamatban, sem a környezetében. Ez elérhető egyrészt a berendezés specifikációjában rögzített feladatok és követelmények maradéktalan teljesítésével, másrészt a vezérelt folyamat ismeretében meghatározott un. biztonsági stabil állapotok valamelyikének kivezérlésével is, amikor a specifikált funkciók nem, vagy nem teljes egészében hajtódnak végre, de a veszélyes állapot kialakulása a folyamat gátlásával valósul meg.

Rendelkezésre nem állás (Q): Annak valószínűsége, hogy a berendezés egy adott időpontban nem képes a specifikációjában rögzített összes feladat és követelmény maradéktalan teljesítésére.

Veszélyeztetettség - biztonság hiánya (US): Annak valószínűsége, hogy a berendezés egy adott időpontban a vezérelt folyamatban vagy a környezetben veszélyes állapotot alakít ki.

$$US(X) = \Psi(\lambda_1, \ldots, \lambda_n), \qquad \textbf{(7-2.)}$$

$$US(X) = \Psi(0, \ldots, 0) = 0 \qquad \textbf{(7-3.)}$$

és

$$\Psi(\lambda_1, \ldots \lambda_i, \ldots, \lambda_n) \le \Psi(\lambda_1, \ldots \lambda_i^{'}, \ldots, \lambda_n) \text{ ha } \lambda_i \le \lambda_i^{'} \qquad \textbf{(7-4.)}$$

Noha a vasúti biztosítóberendezési technikában alkalmazott vizsgálati eljárások a rendszer biztonságosságát vizsgálják (tehát elfogadják hibareakcióként a biztonsági állapot bekövetkezését is), két ok miatt egyre inkább előtérbe kerül a rendelkezésre állás vizsgálata is (akár tapasztalati úton is): egyrészt figyelembe kell venni, hogy a biztosítóberendezés biztonsági állapota esetén emberek irányítják, tartják fent a forgalmat, és ez az emberi hibák bekövetkezésének veszélyét rejti magában, másrészt figyelembe kell venni a forgalom leállásából/lassulásából származó gazdasági hátrányokat is.

A valószínűségi alapú berendezés-vizsgálatok (biztonságosság és rendelkezésre állás) számára szükséges elfogadási határértékek meghatározása az alábbi módokon történhet [Szabó és Tarnai, 1999]:

A biztonságosság esetében:

1. A még elfogadható kockázati szintből való származtatással.
2. Más iparágban, más területen alkalmazott határértékek adaptálásával.
3. Már létező, hosszabb ideje működő berendezések biztonsági szintjének meghatározása segítségével.
4. A már üzemelő berendezések nem valószínűségi alapú vizsgálati módszereiből származtatással.

A rendelkezésre állás esetében:

1. Az üzemeltető számára még tolerálható gazdasági hátrányok meghatározásával.
2. Már létező, hosszabb ideje működő berendezések rendelkezésre állási szintjének meghatározása segítségével.

7.3.2 Az elfogadható kockázati szintből származtatás módszere

A módszer mögötti alapgondolat az, hogy az életnek is van alapkockázata (bármelyik pillanatban történhet velünk valami számunkra nem kedvező), ezért illúzió volna megkövetelni a vasúti közlekedéstől az abszolút kockázatmentességet. Természetesen igen nehéz definiálni a társadalmilag még tolerálható kockázat szintjét, ugyanakkor az új európai vasúti szabványok (EN50126, 129) erre tesznek kísérletet.

A szabványban bemutatott példaszámítás a 7-1 táblázatban található.

Amint a táblázatból látható, az EN50129 szabvány nem rendszerszinten javasolja megállapítani az elfogadható kockázati ráta értékét, hanem származtatás útján a rendszer-elemek szintjén. A számítás problematikája az általános volta: Feltételezve, hogy 100 elem van alrendszerenként… stb. , vagyis nem kínál módszertant konkrét rendszerek esetén történő alkalmazásra.

A másik probléma a mértékegység-választásban rejlik: A biztosítóberendezések veszélyes meghibásodásai nem minden esetben vezetnek balesethez vagy katasztrófához, pusztán csak annak lehetőségét teremtik meg, és csak egyéb feltételek teljesülése esetén következik be a baleset (a legegyszerűbb példa: egy biztosítóberendezési veszélyes meghibásodás csak vonatmozgás esetén okozhat egyáltalán valamilyen balesetet, tehát a vonatmozgás mindenképpen szükséges feltétel). Ennek okán a veszélyes meghibásodások gyakorisága nem ad megfelelő információt a berendezésről: lehet, hogy az egyik berendezésben évente száz veszélyes meghibásodás lép fel, de mindegyik csak néhány másodpercig vagy percig áll fenn, míg a másik berendezésben a veszélyes meghibásodások gyakorisága jóval kisebb, pl. néhány évente, de időtartamuk jóval hosszabb, órák, esetleg napok. A második berendezésnél nagyobb valószínűséggel következik be

veszélyes baleset, mint az elsőnél, és mégis a megbízhatósági paramétere (a rossz választás miatt) jobb. (A példában az adatok torzítottak, egy valós biztosítóberendezésben nem következik be még néhány veszélyes hiba sem évente.) A gyakoriság-paraméter alkalmazása ott előnyös, ahol az állapot fennállás időtartama mellékes. A szabvány kiindulási adata, miszerint évente egy baleset tolerálható, jól mutatja ezt: a baleset bekövetkezésekor érdektelen annak fennállási időtartama. Ugyanakkor az olyan esetekben, amikor a vizsgált esemény bekövetkezése csak a lehetőséget teremti meg egy magasabb szintű esemény bekövetkezéséhez, a bekövetkezési valószínűség használata indokolt.

7-1. táblázat: Valószínűségi határértékek származtatása

FELTÉTELEZÉS	***ELTŰRHETŐ VESZÉLYES MEGHIBÁSODÁSI RÁTA (1/ÓRA)***
Veszélyes műszaki meghibásodásból eredő nagyobb vasúti baleseti ráta tűrhetőnek mondható: évenként 1, Európa egészében.	10^{-4} Az egész európai vasúti rendszer műszaki létesítményei tekintetében.
Feltételezhetően 10-ből 1 veszélyes műszaki meghibásodás vezet nagyobb balesethez.	10^{-3} Az egész európai vasúti rendszer műszaki létesítményei tekintetében.
Feltételezhetően 10-ből 1 veszélyes meghibásodás a biztosítóberendezés meghibásodásából ered.	10^{-4} Az európai vasúti biztosítóberendezés-rendszer létesítményei egészének tekintetében.
A bizonytalanságok és toleranciák érdekében 10:1 biztonsági ráhagyás feltételezése.	10^{-5} Az európai vasúti biztosítóberendezés-rendszer létesítményei egészének tekintetében.
Feltételezhetően 1000 teljes biztosítóberendezés-rendszer van Európa egészében.	10^{-8} Teljes biztosítóberendezés-rendszerenként (pl. nagyobb forgalmi vonal vagy terület).
Feltételezve, hogy 10 alrendszer van teljes biztosítóberendezés-rendszerenként.	10^{-9} Teljes biztosítóberendezés-alrendszerenként (pl. nagy térközbiztosítás).
Feltételezve, hogy 100 rendszerelem van biztosítóberendezés-alrendszerenként.	10^{-11} Rendszerelemenként (pl. vonatérzékelés, váltóállítás stb.).
Feltételezve, hogy 1:100 az arány minden egyes biztonsági szintérték között.	10^{-11} (4-es biztonsági szint), 10^{-9} (3-as biztonsági szint), 10^{-7} (2-es biztonsági szint), 10^{-5} (1-es biztonsági szint), Rendszerelemenként.

7.3.3 A már minősített, üzemelő rendszerek megbízhatósági szintjeinek meghatározása

Ebben az esetben egyedi, már működő biztosítóberendezések jellemzőit lehetne meghatározni valószínűségi alapú vizsgálati módszerekkel. Ezeknél az üzemelő berendezéseknél a valós meghibásodási ráták az üzemviteli tapasztalatok útján meghatározhatóak, a valószínűségi modellek paraméterei pontosan megadhatóak. Ugyanakkor felmerül a kérdés, hogy egy (néhány) működő berendezés vizsgálata alapján vajon a valós határértékeket állapíthatjuk-e meg, vagy létezhetnek olyan berendezések is, amelyek a már vizsgált berendezéseknél rosszabb rendszerjellemzőket mutatnak.

Ezek alapján a konkrét berendezések vizsgálatánál célszerűbbnek látszik a berendezések vizsgálati módszereinek elemzése, és a vizsgálati módszerek alapján elfogadási határértékek megállapítása.

7.3.4 A már üzemelő berendezések nem valószínűségi alapú vizsgálati módszereiből származtatással

7.3.4.1 Általános megfontolások

A determinisztikus alapú minősítési eljárás alapján elfogadott rendszer esetén a biztonság szintje 1-re adódik, mivel e rendszer működése során egyetlen pillanatra sem (illetve csak a reakciók végrehajtásához szükséges, a berendezés élettartama és a rendszerkomponensek szempontjából igen rövid ideig) állhat fent olyan meghibásodás vagy meghibásodási kombináció, amelynek a hatása veszélyes lehet. Ugyanakkor az így minősített rendszerek rendelkezésre állásáról nem szerezhetünk információkat, mivel a minősítési eljárás elfogadja, ha egy meghibásodásra a rendszer a definiált biztonsági állapot felvételével reagál, és nem támaszt követelményeket az ilyen reakció gyakoriságára, vagy pl. a biztonsági állapot maximális fennállási idejére vonatkozóan.

A sokkal realisztikusabb, MÜ8004 alapú vizsgálatnál a második hiba fellépési időn belül megengedett egy meghibásodás fennállása, amelynek egyedi hatása a vizsgálat szerint nem lehet veszélyeztető, és második (vagy következő) meghibásodással is csak akkor számol az eljárás, ha ez az első meghibásodás a második hiba fellépési időn belül nem detektálódik és nem okoz biztonsági állapotot, vagy nem kerül elhárításra.

A módszer mögötti gondolat az, hogy a rendszer elemei meghibásodási valószínűségeinek ismeretében meghatározható egy olyan intervallum (ez lesz a második hiba fellépési idő), amelyen belül annak a valószínűsége, hogy egynél több meghibásodás lép fel, “elenyésző”, ugyanakkor számunkra mégis számszerűsíthető értéket képvisel.

Tekintsünk egy általános rendszert *n* komponenssel. Tételezzük fel, hogy az egyes komponensek azonos meghibásodási rátával (λ) rendelkeznek. Amennyiben ez nem lenne igaz,

$$\lambda = \max_{i=1..n}(\lambda_i) \qquad \textbf{(7-5.)}$$

kitétellel élhetünk, így az elemzésünk által szolgáltatott értéknél a vizsgált rendszer biztonságosabb. A továbbiakban már ezzel az egységenként azonos meghibásodási rátával számolunk.

A következő részekben rendre megvizsgáljuk a rendszer viselkedését, először azzal a feltételezéssel, hogy két egyidejű meghibásodás veszélyes állapotot eredményezhet, majd azzal a feltételezéssel, hogy csak három egyidejű meghibásodás vezethet veszélyes állapothoz, majd végül azzal a feltételezéssel, hogy a rendszerben csak háromnál több egyidejű meghibásodás okozhat veszélyes állapotot.

7.3.4.2 Két egyidejű meghibásodás veszélyes állapotot eredményez

A második hiba fellépési idő számításánál azt feltételezzük, hogy a rendszer összes komponense részt vehet veszélyes meghibásodási állapot előállításában, így mindegyiket figyelembe kell venni a második hiba fellépési idő számításához. Amennyiben az *n* komponens közül tetszőleges kettő egyidejű hibás állapota kiválthatja a veszélyes állapotot:

$$T_S = \frac{1}{1000\sum_{i=1}^{n}\lambda_i} = \frac{1}{1000 \cdot n \cdot \lambda} \qquad \textbf{(7-6.)}$$

A detektálási, hibahatás semlegesítési (pl. biztonsági állapot felvétele vagy javítás) időtartamának maximuma egyenlő a második hiba fellépési idővel, az ebből számolt ráta pedig a második hiba fellépési idő reciprokával.

$$\mu = \frac{1}{T_S} = 1000\sum_{i=1}^{n}\lambda_i = 1000n\lambda \qquad \textbf{(7-7.)}$$

A rendszert úgy minősítették, hogy figyelmen kívül hagyták egy esetleges meghibásodás után, a második hiba fellépési időn belül bekövetkező további meghibásodások és az első meghibásodás együttes hatását (pl. azért, mert az első meghibásodás a második hiba fellépési idő letelte előtt biztosan detektálásra került). Feltételezésünk szerint ezek a nem vizsgált, többszörös hibák veszélyes állapotot hoznak létre (worst case feltételezés).

Egy komponens meghibásodásának valószínűsége az idő függvényében folyamatosan tesztelt, javított komponens-modellel írható le.

$$P_{első\ meghibásodás}(t) = \frac{\lambda}{\lambda+\mu}\cdot\left(1-e^{-(\lambda+\mu)t}\right) = \frac{\lambda}{(1000n+1)\lambda}\cdot\left(1-e^{-(1000n+1)\lambda t}\right) \qquad \textbf{(7-8.)}$$

amely relatíve gyorsan eléri az állandósult állapotbeli értéket:

$$P_{első\ meghibásodás} = \frac{\lambda}{(1000n+1)\lambda} \qquad \textbf{(7-9.)}$$

Annak a valószínűsége, hogy a rendszer *n* komponense közül egy és csakis egy hibásodik meg egy adott időpillanatban:

$$\begin{aligned} &P(csak\ egy\ kiválasztott\ komponens\ hibás) = \\ &P(a\ kiválasztott\ komponens\ hibás)\cdot P(n-1\ komponens\ jó) \end{aligned} \qquad \textbf{(7-10.)}$$

$$P\ (csak\ egy\ kiválasztott\ komponens\ hibás) = \frac{\lambda}{(1000n+1)\lambda}\cdot\left(1-\frac{\lambda}{(1000n+1)\lambda}\right)^{(n-1)} \qquad \textbf{(7-11.)}$$

A rendszerben bármelyik komponens meghibásodhat, és a csakis egy hibát tartalmazó események egymást kizárják, így annak a valószínűsége, hogy a rendszerben éppen egy komponens hibás, az alábbi:

$$P\ (csak\ egy\ komponens\ hibás) = n\cdot\frac{\lambda}{(1000n+1)\lambda}\cdot\left(1-\frac{\lambda}{(1000n+1)\lambda}\right)^{(n-1)} \qquad \textbf{(7-12.)}$$

A második meghibásodás már nem ugyanolyan valószínűség-idő függvénnyel írható le, mint az első, mert most már a meghibásodás bekövetkeztekor a javítás

érdektelen, hiszen elértük a veszélyes állapotot. Mivel ez a második meghibásodás még nem állt fenn, amikor az első bekövetkezett, valamint a monoton növekvő jellege miatt a $t=T_s$ időpillanat a legkritikusabb, az alábbiakat mondhatjuk:

$$P_{második\ hiba}\left(t=T_S\right)=1-e^{-\frac{\lambda}{1000n\lambda}}=P_2 \qquad \textbf{(7-13.)}$$

Ennél a pontnál érdektelen számunkra, hogy a rendszerben az első meghibásodás mellé egy és csakis egy, vagy legalább egy meghibásodás lépett fel, így nem kell figyelembe vennünk *n-2* komponens működőképességének valószínűségét. Mivel most az *n-1*, az első meghibásodás pillanatában még működőképes komponens egyedi meghibásodása nem egymást kizáró esemény, annak a valószínűsége, hogy még legalább egy meghibásodás bekövetkezik a $t=T_s$ időpillanatban:

$$P(újabb\ komponens\ hiba)=1-\left(1-P_{második\ hiba}\left(t=T_s\right)\right)^{(n-1)}=$$
$$1-\left(1-\left(1-e^{-\frac{\lambda}{1000\cdot n\cdot\lambda}}\right)\right)^{(n-1)}=1-e^{-\frac{(n-1)}{1000\cdot n}} \qquad \textbf{(7-14.)}$$

Az eredmények felhasználásával annak a maximális valószínűsége, hogy egy fellépett meghibásodás után a második hiba fellépési időtartam alatt újabb meghibásodás (vagy meghibásodások) lépnek fel:

$$P_{\max}=P(csak\ egy\ komponens\ hibás)\cdot P(újabb\ komponens\ hiba)=$$
$$=n\cdot\frac{\lambda}{(1000n+1)\lambda}\cdot\left(1-\frac{\lambda}{(1000n+1)\lambda}\right)^{(n-1)}\cdot\left(1-e^{-\frac{n-1}{1000\cdot n}}\right)= \qquad \textbf{(7-15.)}$$
$$=\frac{n}{1000n+1}\cdot\left(1-\frac{1}{1000n+1}\right)^{(n-1)}\cdot\left(1-e^{-\frac{n-1}{1000\cdot n}}\right)$$

Úgy tűnik, hogy a kapott összefüggés független a komponensek meghibásodási rátáitól, és bármilyen jóságú komponensekből azonos biztonsági szintű rendszert lehetne építeni. Ez valóban igaz, azonban nem szabad azt elfelejteni, hogy az egyes meghibásodásokat a definiált második hiba fellépési időn belül kell felfedni, amely nagy meghibásodási rátájú komponensek esetén igen rövid időtartam is lehet, és így a feltétel betartása nagy ráfordításokat igényel.

Vizsgáljuk meg a fenti eredmény számszerű értékét *n* szélső értékeinél. Az így kapott eredmény lesz felhasználható valószínűségi vizsgálatok határértékeként.

- $n=2$ esete:

$$P_{\max}=\frac{2}{2001}\cdot\left(1-\frac{1}{2001}\right)^{(2-1)}\cdot\left(1-e^{-\frac{2-1}{2000}}\right)= \qquad \textbf{(7-16.)}$$
$$=\left(4.99\cdot10^{-7}-2.4962\cdot10^{-10}\right)\cdot0.6323=4{,}9912\cdot10^{-7}$$

- $n \to \infty$ esete (itt határérték-számítást alkalmazunk):

$$P_{\max} = \lim_{n\to\infty}\left[\frac{n}{1000n+1}\cdot\left(1-\frac{1}{1000n+1}\right)^{(n-1)}\cdot\left(1-e^{-\frac{n-1}{1000\cdot n}}\right)\right]= \\ = \lim_{n\to\infty}\left[\frac{n}{1000n+1}\right]\cdot\lim_{n\to\infty}\left[\left(1-\frac{1}{1000n+1}\right)^{(n-1)}\right]\cdot\lim_{n\to\infty}\left[\left(1-e^{-\frac{n-1}{1000\cdot n}}\right)\right] \quad \textbf{(7-17.)}$$

A könnyebb átláthatóság és ellenőrizhetőség érdekében a határérték-számítást három rész-határérték kiszámítására vezetjük vissza. Az egyes rész-határértékek értéke:

Az első rész-határérték:

$$\lim_{n\to\infty}\left[\frac{n}{1000n+1}\right]=\frac{1}{1000}=10^{-3} \quad \textbf{(7-18.)}$$

A második rész-határérték:

$$\lim_{n\to\infty}\left[\left(1-\frac{1}{1000n+1}\right)^{(n-1)}\right]=\lim_{n\to\infty}\left[\frac{\left(\frac{\left(1-\frac{1}{1000n+1}\right)^{(1000n+1)}}{1-\frac{1}{1000n+1}}\right)^{\frac{1}{1000}}}{\left(1-\frac{1}{1000n+1}\right)}\right]= \\ = \frac{\left(\frac{\lim_{n\to\infty}\left[\left(1-\frac{1}{1000n+1}\right)^{(1000n+1)}\right]}{\lim_{n\to\infty}\left[1-\frac{1}{1000n+1}\right]}\right)^{\frac{1}{1000}}}{\lim_{n\to\infty}\left[1-\frac{1}{1000n+1}\right]}=\frac{\left(\frac{\frac{1}{e}}{1}\right)^{\frac{1}{1000}}}{1}=\left(\frac{1}{e}\right)^{\frac{1}{1000}}=0.9990004999 \quad \textbf{(7-19.)}$$

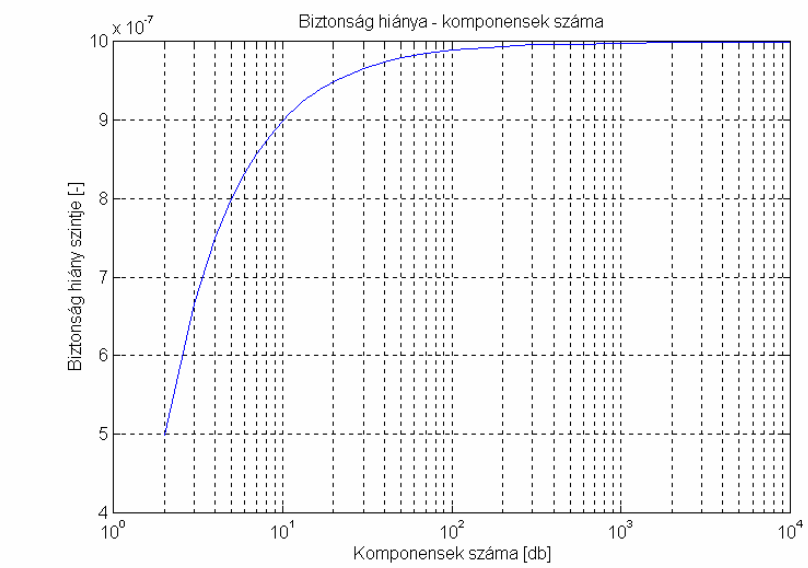
A harmadik rész-határérték:

$$\lim_{n\to\infty}\left[1-e^{-\frac{n-1}{1000\cdot n}}\right]=1-e^{-1\cdot\lim_{n\to\infty}\left[\frac{n-1}{1000\cdot n}\right]}=1-e^{-\frac{1}{1000}} \quad \textbf{(7-20.)}$$

A fenti határértékek alkalmazásával a veszélyes állapot bekövetkezésének maximális valószínűsége:

$$P_{\max} = 10^{-3} \cdot 0.9990004999 \cdot \left(1 - e^{-0.001}\right) = 9.985 \cdot 10^{-7} \qquad \textbf{(7-21.)}$$

A 7-3. ábra bemutatja a veszélyes hiba bekövetkezési valószínűségének függését a rendszerben alkalmazott komponensek számától, amennyiben két egyidejű meghibásodás veszélyes lehet.



7-3. ábra: Veszélyes hiba bekövetkezési valószínűségének függése a rendszerben alkalmazott komponensek számától, amennyiben két egyidejű meghibásodás veszélyes lehet

Amint az várható volt, a komponensek számának növekedésével növekszik a rendszer biztonság hiányának szintje. Annak bizonyítására, hogy a keresett valószínűség értéke *n* növelésével monoton növekszik a fenti értékig, vizsgáljuk meg, hogy a függvény deriváltja felveszi-e a nulla értéket véges *n* mellett. A vizsgálat részletesen a 3. mellékletben található

Mivel a derivált értéke *n*=2-től (a soros rendszer miatt *n* minimális értéke 2), pozitív így $\left(P_{felső}\right)' = 0$ csak $n \to \infty$ esetén teljesül, vagyis az elemzésünkben *n* minden határon túl való növelésével a maximális biztonság hiány szintet határoztuk meg.

Összefoglalva: Bizonyítottuk, hogy a MŰ8004 eljárás alapján történő rendszer-alkalmassági vizsgálattal olyan rendszerek elfogadhatónak minősíthetőek, amelyek $9.985 \cdot 10^{-7}$-es biztonsági hiány szintet valósítanak meg. Azok a rendszerek, amelyek az előbbi értéknél kevésbé biztonságosak, már nem felelnek meg a MÜ8004 vizsgálati, ill. elfogadási elveknek. Az így meghatározott számérték tehát

felhasználható a valószínűségi alapú biztosítóberendezési vizsgálatokhoz elfogadási határérték céljára.

A meghatározott valószínűség segítségével megpróbálhatjuk megállapítani azoknak a rendszerszintű meghibásodásoknak a gyakoriságát is, amelyek veszélyes állapotot okozhatnak.

Ehhez definiálnunk kell a biztonsági szint hiányának valószínűségét:

$$P(US) = \\ = \frac{Vizsgálati\ idő\ alatt\ bekövetkező, veszélyes\ állapotot\ eredményező\ meghibásodások\ időtartama}{Vizsgálati\ idő} = \\ = \frac{Vizsgálati\ idő * Meghibásodási\ ráta(rendszer) * Elhárítási\ idő}{Vizsgálati\ idő} = \lambda_{rendszer} \cdot T_{javítás}$$

(7-22.)

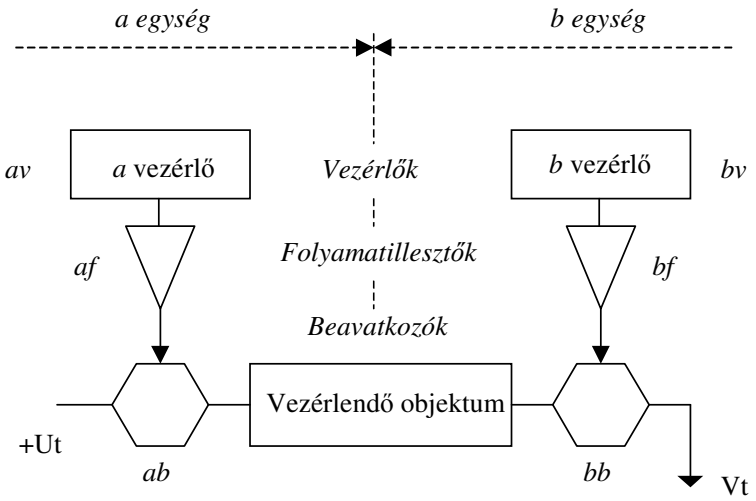
$$\lambda_{rendszer} = \frac{6.31 \cdot 10^{-7}}{T_{javítás}} = \frac{6.31 \cdot 10^{-7}}{T_S}$$

(7-23.)

Amint azt korábban láthattuk, P(US) független volt a rendszerben alkalmazott komponensek meghibásodási rátájától. Mivel azonban a második hiba fellépési idő tartalmazza a rátát, így a rendszer meghibásodási rátája csak a komponensek meghibásodási rátájának ismeretében volna számítható.

7.3.4.3 Példa a módszer alkalmazására

Tekintsünk egy vasúti vezérlőrendszert, amely kétsarkúan vezérel egy objektumot. Az aktív állapot kivezérlése csak akkor lehetséges, ha azt az *a* és a *b* egység is vezérli (7-4. ábra).



7-4. ábra: A példarendszer vezérlési vázlata

A rendszerben veszélyes állapot a vezérlőberendezés miatt akkor áll elő, ha a vezérlendő objektum lekapcsolása meghibásodás miatt lehetetlenné válik. A mintarendszerben ez az alábbi esetekben következhet be (7-2. táblázat):

7-2. táblázat: Példarendszer minimális vágatai

Sorszám	A veszélyes állapothoz minimálisan szükséges meghibásodások		Sorszám	A veszélyes állapothoz minimálisan szükséges meghibásodások
1.	av, bv		6.	af, bb
2.	av, bf		7.	ab, bv
3.	av, bb		8.	ab, bf
4.	af, bv		9.	ab, bb
5.	af, bf			

A vezérlőrendszerben található elemek száma 6. Tételezzük fel, hogy az elemek meghibásodási rátája egységesen 0,1/év, tehát egy-egy elemtípust tekintve, átlagosan tíz évente következik be egy meghibásodás. Noha nem bármely két elem meghibásodása vezet a veszélyes állapothoz, a második hiba fellépési idő:

$$T_S = \frac{1}{1000\sum_{i=1}^{n}\lambda_i} = \frac{1}{1000 \cdot 6 \cdot 0.1} = 1.66 \cdot 10^{-3}\ év = 14.6\ óra \qquad \textbf{(7-24.)}$$

és

$$P(US) \approx 4.25 \cdot 10^{-7} \qquad \textbf{(7-25.)}$$

Tehát amennyiben ezen az időn belül biztosítható a meghibásodás detektálása és javítása (pl. automatikus, ciklikus tesztekkel és állandóan rendelkezésre álló karbantartó szolgálattal), akkor a berendezés egy meghibásodás fellépése esetén is tovább üzemelhet. Ellenkező esetben biztosítani kell a fenti időtartamon belüli biztonsági állapot felvételt.

7.3.4.4 Három egyidejű meghibásodás veszélyes állapotot eredményez

A MÜ8004 szerint, amennyiben az *n* komponens közül csak bármely három tetszőleges komponens együttes meghibásodása vezet veszélyes állapothoz, a fellépett hiba detektálására és a hibahatás elhárítására (javítás vagy biztonsági állapot felvétel) rendelkezésre álló idő (nevezzük továbbra is második hiba fellépési időnek, bár itt egy kicsit más tartalmat nyer a fogalom):

$$T_S = \frac{2}{\sum_{i=1}^{n}\lambda_i} = \frac{1}{0.5 \cdot n \cdot \lambda} \qquad \textbf{(7-26.)}$$

az ebből számolt ráta pedig:

$$\mu = \frac{1}{T_S} = 0.5 \cdot \sum_{i=1}^{n}\lambda_i = 0.5 \cdot n \cdot \lambda \qquad \textbf{(7-27.)}$$

Ennél a rendszerminősítésnél is figyelmen kívül hagyták egy esetleges meghibásodás után, a második hiba fellépési időn belül bekövetkező további két meghibásodás és az első meghibásodás együttes hatását (pl. azért, mert az első meghibásodás a második hiba fellépési idő letelte előtt biztosan detektálásra került). Továbbra is azt feltételezzük, hogy a nem vizsgált, többszörös hibák veszélyes állapotot hoznak létre (worst case feltételezés).

Egy komponens meghibásodásának valószínűsége az idő függvényében folyamatosan tesztelt, javított komponens-modellel írható le.

$$P_{első\ meghibásodás}(t) = \frac{\lambda}{\lambda+\mu}\cdot\left(1-e^{-(\lambda+\mu)t}\right) = \frac{\lambda}{(0.5n+1)\lambda}\cdot\left(1-e^{-(0.5n+1)\lambda t}\right) \qquad \textbf{(7-28.)}$$

amely egy idő után eléri az állandósult állapotbeli értéket.

$$P_{első\ meghibásodás} = \frac{\lambda}{(0.5n+1)\lambda} \qquad \textbf{(7-29.)}$$

Annak a valószínűsége, hogy a rendszer *n* komponense közül egy és csakis egy hibásodik meg egy adott időpillanatban:

$$\begin{aligned} &P(csak\ egy\ kiválasztott\ komponens\ hibás) = \\ &P(a\ kiválasztott\ komponens\ hibás)\cdot P(n-1\ komponens\ jó) \end{aligned} \qquad \textbf{(7-30.)}$$

$$P\ (csak\ egy\ kiválasztott\ komponens\ hibás) = \frac{\lambda}{(0.5n+1)\lambda}\cdot\left(1-\frac{\lambda}{(0.5n+1)\lambda}\right)^{(n-1)} \qquad \textbf{(7-31.)}$$

A rendszerben bármelyik komponens meghibásodhat, és a csakis egy hibát tartalmazó események egymást kizárják, így annak a valószínűsége, hogy a rendszerben éppen egy komponens hibás, az alábbi:

$$P\ (csak\ egy\ komponens\ hibás) = n\cdot\frac{\lambda}{(0.5n+1)\lambda}\cdot\left(1-\frac{\lambda}{(0.5n+1)\lambda}\right)^{(n-1)} \qquad \textbf{(7-32.)}$$

A második, majd harmadik meghibásodás ugyanolyan valószínűség-idő függvénnyel írható le, mint az első. Mivel ezek a meghibásodások még nem álltak fenn, amikor az első bekövetkezett, valamint a monoton növekvő jellege miatt a $t=T_s$ időpillanat a legkritikusabb, az alábbiakat mondhatjuk:

$$P_{második\ hiba}(t=T_S) = 1-e^{-\frac{\lambda}{0.5\cdot n\cdot\lambda}} = P_3 \qquad \textbf{(7-33.)}$$

Ellentétben a két meghibásodást tárgyaló esetnél, most azt vizsgáljuk, mekkora a valószínűsége annak, hogy két komponens hibásodik meg az első hiba után még működőképesek közül, és az összes többi továbbra is működőképes marad.

$$P(két\ újabb\ komponens\ hiba) = P_3^2\cdot(1-P_3)^{(n-3)} \qquad \textbf{(7-34.)}$$

Azonban a rendszerben bármely két, korábban működőképes elem meghibásodása problémát okozhat, ezek mind egymást kizáró események, így:

$$P(két\ újabb, általános\ komponens\ hiba) = \binom{n-1}{2}\cdot\left(P_3^2\cdot(1-P_3)^{(n-3)}\right) \qquad \textbf{(7-35.)}$$

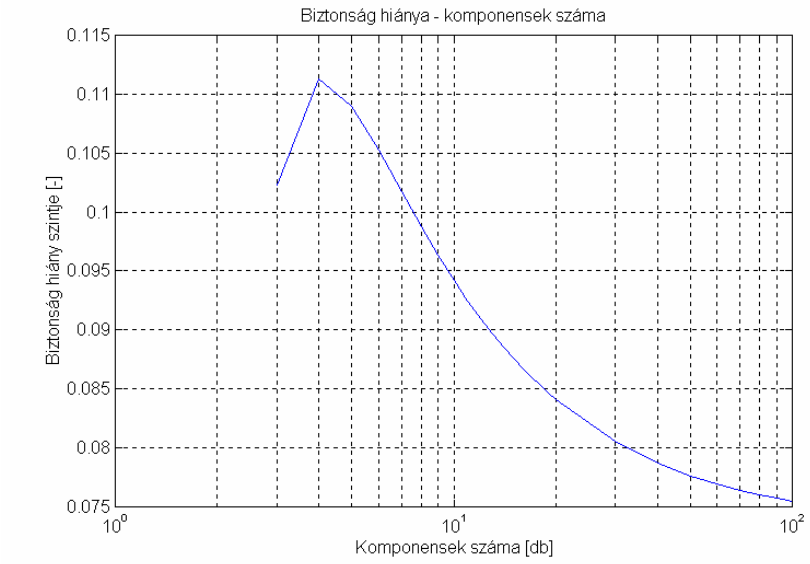
Az eredmények felhasználásával annak a maximális valószínűsége, hogy egy fellépett meghibásodás után a második hiba fellépési időtartam alatt újabb két meghibásodás lép fel:

$$P_{\max} = P(csak\ egy\ komponens\ hibás) \cdot P(két\ újabb, általános\ komponens\ hiba) =$$

$$= \frac{n\lambda}{(0.5n+1)\lambda} \cdot \left(1 - \frac{\lambda}{(0.5n+1)\lambda}\right)^{(n-1)} \cdot \binom{n-1}{2} \cdot \left(1 - e^{-\frac{\lambda}{0.5n\lambda}}\right)^2 \cdot e^{-\frac{(n-3)\lambda}{0.5n\lambda}} = \qquad \textbf{(7-36.)}$$

$$= \frac{n}{0.5n+1} \cdot \left(1 - \frac{1}{0.5n+1}\right)^{(n-1)} \cdot \binom{n-1}{2} \cdot \left(1 - e^{-\frac{1}{0.5n}}\right)^2 \cdot e^{-\frac{n-3}{0.5n}}$$

A 7-5. ábra a veszélyes hiba bekövetkezési valószínűségének függését mutatja a rendszerben alkalmazott komponensek számától, amennyiben három egyidejű meghibásodás veszélyes lehet. A grafikon értékelésénél megállapíthatjuk, hogy a második hiba fellépési idő jelentős növelése rossz hatással volt a rendszerre, annak ellenére, hogy nem elégséges két egyidejű meghibásodás fennállása a veszélyes állapot bekövetkezéséhez. A nagy megbízhatóságú rendszerekkel kapcsolatos tapasztalatokkal összehasonlítva az eredményeket, a biztonság hiány valószínűsége a szokásosnál jelentősen nagyobbra adódik, így ennek az értéknek a valószínűségi vizsgálatokhoz való alkalmazása nem javasolható. Ugyanakkor meg kell jegyezni, hogy a biztosítóberendezési gyakorlatban ez az eset, amelyben két egyidejű meghibásodás nem okozhat veszélyes állapotot, nagyon ritka, így ez a tény sem indokolja az ebből az esetből származó eredmények alkalmazását.



7-5. ábra: Veszélyes hiba bekövetkezési valószínűségének függése a rendszerben alkalmazott komponensek számától, amennyiben három egyidejű meghibásodás veszélyes lehet

7.3.4.5 Háromnál több egyidejű meghibásodás eredményez veszélyes állapotot

Amennyiben csak négy egyidejű meghibásodás lehet veszélyes, és az elemek meghibásodási rátáinak összege kisebb, mint $2*10^{-4}$ h^{-1}, a MÜ8004 nem ír elő detektálási, ill. hibasemlegesítési követelményt. Amennyiben az ilyen berendezésekben valóban nem valósítanak meg hibadetektálást, és csakugyan van olyan négyes hibakombináció, amely veszélyes állapothoz vezet, akkor egy adott időpontban a veszélyes állapot bekövetkezésének valószínűsége (feltételezve, hogy a rendszert felépítő *n* darab komponens meghibásodási rátáinak összege pontosan a MÜ8004 által megengedett maximális értéket adja ki):

$$P_{\text{négy vagy több meghibásodás}}(t) =$$

$$= 1 - \left(\left(e^{-\frac{2\cdot10^{-4}}{n}\cdot t} \right)^n + n \cdot \left(1 - e^{-\frac{2\cdot10^{-4}}{n}\cdot t} \right) \cdot \left(e^{-\frac{2\cdot10^{-4}}{n}\cdot t} \right)^{(n-1)} + \right.$$

$$\left. + \left(\binom{n}{2} \cdot \left(1 - e^{-\frac{2\cdot10^{-4}}{n}\cdot t} \right)^2 \cdot \left(e^{-\frac{2\cdot10^{-4}}{n}\cdot t} \right)^{(n-2)} \right) + \left(\binom{n}{3} \cdot \left(1 - e^{-\frac{2\cdot10^{-4}}{n}\cdot t} \right)^3 \cdot \left(e^{-\frac{2\cdot10^{-4}}{n}\cdot t} \right)^{(n-3)} \right) \right)$$

(7-37.)

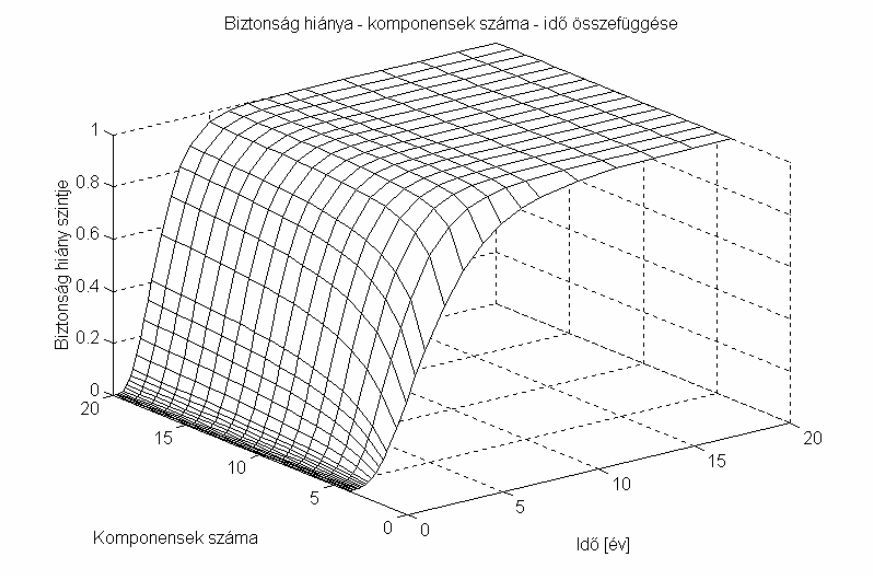
ahol a második, negatív előjelű tag megadja annak a valószínűségét, hogy egy komponens sem hibás, vagy csak egy, két vagy három komponens hibás, míg a teljes összefüggés ennek a logikai feltételnek a negáltja, vagyis a rendszerben háromnál több meghibásodás lépett fel.

Mivel a rendszerkomponenseknél nincs hibadetektálás, és ennek következtében hibajavítás sem, az egyes komponensek meghibásodási valószínűsége 1-hez tart, és ennek következtében a rendszer által okozott veszélyes állapot bekövetkezési valószínűsége is 1-hez tart. A kérdés csak az, hogy a biztosítóberendezések átlagos élettartamán belül (kb. 50 év) mekkora értékig emelkedik ez a valószínűség.

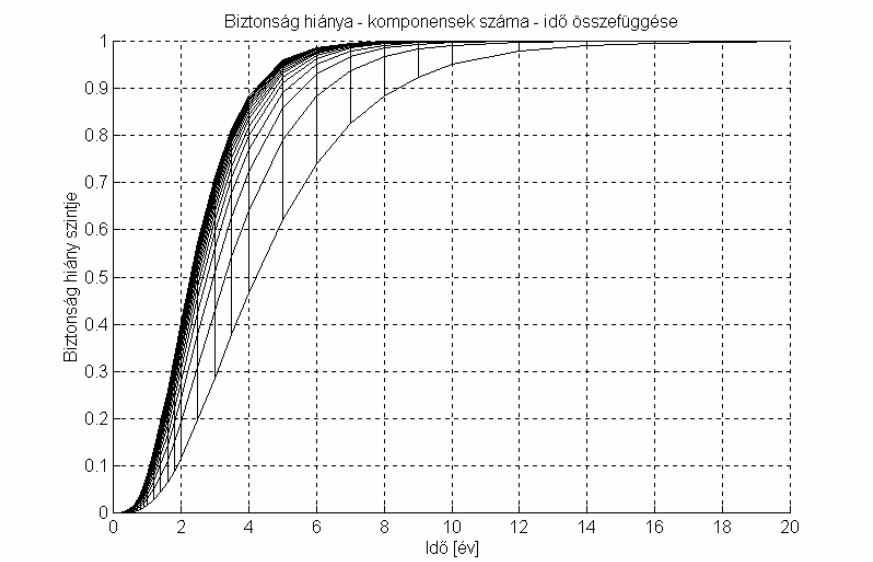
A 7-6. ábra háromdimenziós grafikonként bemutatja a veszélyes hiba bekövetkezési valószínűségének függését a rendszerben alkalmazott komponensek számától és az időtől, amennyiben háromnál több egyidejű meghibásodás lehet csak veszélyes. A 7-7. ábra ugyanezt az összefüggést mutatja kétdimenziós grafikonként, különböző rendszerkomponens-darabszám esetére. Ebben a grafikonban a legalsó görbe 4 komponensből álló rendszerhez tartozik, míg a rendre következő görbék eggyel-eggyel több komponenst tartalmazó rendszert írnak le.

A görbéket vizsgálva megállapítható, hogy a biztosítóberendezések átlagos élettartama alatt az ilyen módon minősített rendszerben a veszélyes állapotot okozó hibakombináció bekövetkezési valószínűsége megközelíti a 100%-ot.

Másképpen fogalmazva: lehetséges olyan vasúti biztosítóberendezési architektúra tervezése, amelyen alapuló berendezésekben csak négy, egy időben fennálló komponens-hiba eredményez veszélyes állapotot, három egyidejű hiba nem akadályozó jellegű, a komponensek meghibásodási rátáinak összege kisebb, mint $2*10^{-4}$ h^{-1}, valamint a rendszerben (pl. egyszerűségi okokból) nem valósítanak meg hibadetektálást. Az ilyen biztosítóberendezés a MÜ8004 szabvány alapján megfelelőnek minősíthető, noha néhány év elteltével a veszélyes hibakombinációk bekövetkezésének valószínűsége közelíti a 100 %-ot.

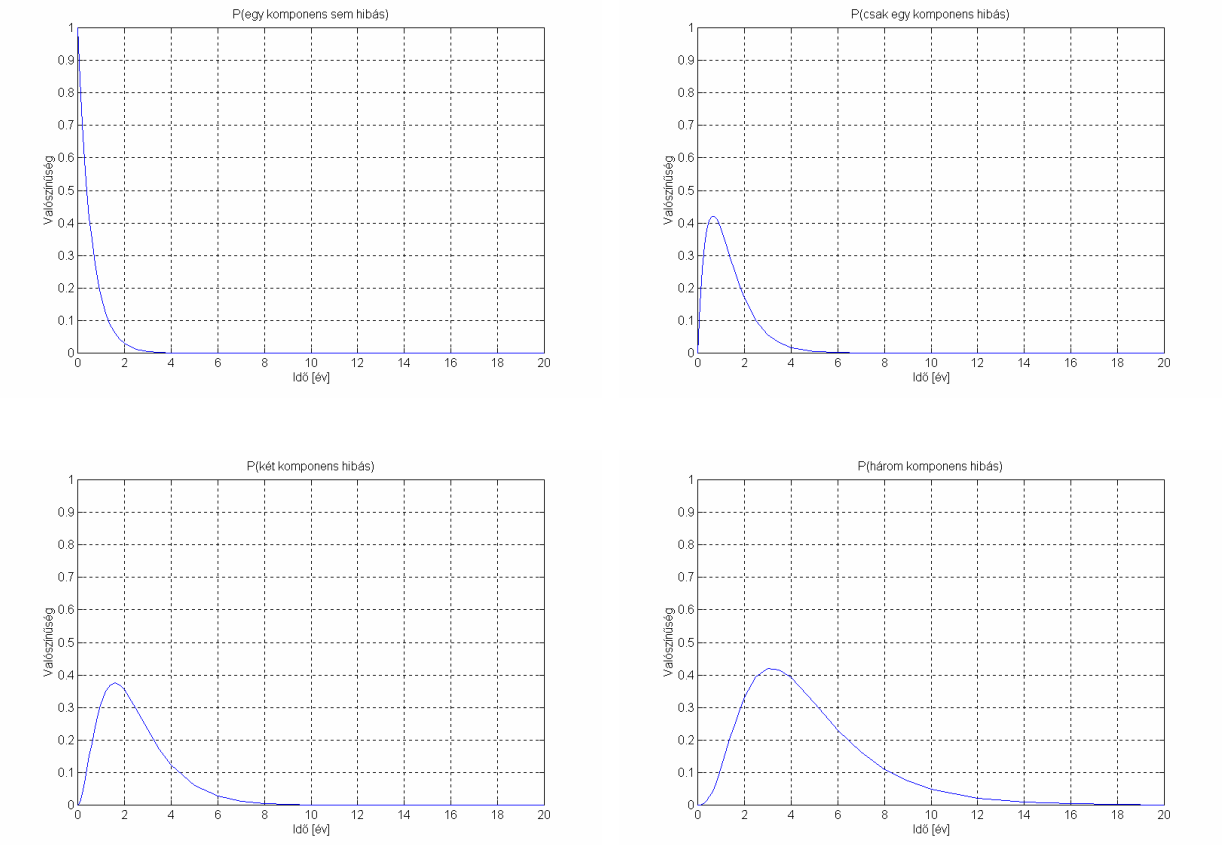


7-6. ábra: Veszélyes hiba bekövetkezési valószínűségének függése a rendszerben alkalmazott komponensek számától és az időtől, amennyiben háromnál több egyidejű meghibásodás lehet csak veszélyes



7-7. ábra: Veszélyes hiba bekövetkezési valószínűségének függése a rendszerben alkalmazott komponensek számától és az időtől, amennyiben háromnál több egyidejű meghibásodás lehet csak veszélyes

A 7-8. ábrán 4 komponens esetére bemutatjuk a 7-38. képletben alkalmazott rész-valószínűségek időfüggését is. A képlet szerint annak a valószínűsége, hogy 4 komponens hibás, számolható a komplementer eseményből is. A komplementer esemény a 4-nél kevesebb meghibásodást tartalmazó állapotok összessége: nem áll fenn a rendszerben meghibásodás, vagy csak 1, 2, 3 meghibásodás áll fenn. A komplementer esemény-tagok valószínűségének alakulását mutatja a 7-8 ábra négy grafikonja.



7-8. ábra: Részvalószínűségek

IV. Tézis: Bebizonyítottam, hogy biztosítóberendezések valószínűségi alapú minősítéséhez tartozó elfogadási határértékre a korábban minősített rendszerek alapján becslés adható.

IV.A: Megállapítottam, hogy a korábban tiszta determinisztikus eljárással vizsgált berendezés alapján sem a valós biztonsági szint, sem a berendezés rendelkezésre állása nem becsülhető.

IV.B: Bebizonyítottam, hogy a korábban MÜ8004 alapú eljárással vizsgált berendezés biztonsági szintjére felső korlát számítható, amennyiben a berendezésben két egyidejű, nem detektált meghibásodás veszélyes állapotot eredményezhet. Megállapítottam, hogy az így nyert érték alkalmas újabb berendezések valószínűségi alapú vizsgálatához elfogadási küszöbértéknek. A felső korlát értéke nem függ a berendezésben alkalmazott elemek megbízhatóságától, csak a vizsgálatnál a második hiba fellépési idő számítására használt képletben alkalmazott segédszorzótól. Javaslatot adtam a felső korlátra, a javasolt valószínűségi határérték 10^{-6}.

IV.C: Megállapítottam, hogy a korábban MÜ8004 alapú eljárással vizsgált berendezés rendelkezésre állási szintje nem becsülhető.

8. ZÁRSZÓ

A disszertáció célja a kockázati alapú rendszermegbízhatóság-kezelés egyes részterületeinek fejlesztése, ezen belül elsősorban a megbízhatóság-elmélet általános technikáinak kiterjesztése egy speciális, napjainkban előtérbe kerülő terület, a hiba-adaptív logikák analízisének céljára.

A kockázati alapú rendszerfejlesztés - noha az alapelvek már jó néhány éve ismertek - csak napjainkban kezd igazán elterjedni az egyes, korábban nem ilyen elvek szerint kezelt alkalmazások kockázatos voltának felismerésével, a kockázati szintek felmérésével. Az elterjedés azonban több problémacsoportot is a felszínre hoz: részben a projektek egyre nagyobb diverzitása újabb és újabb kockázat elemzési és értékelési módszerek kifejlesztését igényli, részben egyre több szakmai és laikus folyamatrészvevőnek (fejlesztőknek, gyártásban részt vevőknek, alkalmazóknak vagy a biztonságkritikus rendszerekkel valamilyen nem rendszeres módon kapcsolatba kerülő személyeknek) kell tudni értelmezni a kockázati alapú megközelítésmódot.

Éppen ezért véleményem szerint nem csak az alkalmazandó módszerek kutatása, fejlesztése a fontos: kiemelt hangsúlyt kell fordítanunk a kockázati alapú megközelítés érthető kommunikálására, a megvalósítható biztonság fogalmának megértetésére, a biztonság eléréséhez szükséges erőforrások nagyságának felismertetésére. Mindezeknek a kérdéseknek társadalmi, de jogi és gazdasági vetületei is vannak, ezért a szakterületen dolgozók feladata a közérthető, az alapoktól induló, a miérteket is részletesen magyarázó publikációk megjelentetése is a részletekre vonatkozó módszerek publikálása mellett.

A disszertációban publikált eredmények alapjául szolgáló kutatás - az előbb említett szakterületi kihívások sokasodása okán - tovább folyik: részben a zárt alakú képletek terén van lehetőség további általánosításra, amelynek eredményeképpen nem csak a rendszerek egy körére, hanem esetlegesen megkötés nélkül bármilyen rendszerre alkalmazhatóak lesznek a közölt technikák. Ugyancsak továbblépés lehetséges az időfüggő modellek (komplex modellek) terén: részben újabb elektronikus viselkedésmódok leírására van lehetőség, részben pedig az elektronikus rendszerekre alkalmazott modellek terjeszthetők ki az elektronika által vezérelt vagy kezelt mechanikai rendszerekre is, ezáltal lehetővé téve a megbízhatóság-analízis határainak kijjebb tolását. És triviális a harmadik továbblépési irány is: a bemutatott elveket, módszereket minél szélesebb körben, ipari munkákban alkalmazni, akár a légiközlekedési irányítórendszerek, akár a repülő objektumok, de akár vasúti vagy közúti rendszerek vonatkozásában is.

És e mellett természetesen tovább kell folytatnunk a biztonságkritikus rendszerekre, a kockázatelemzés és kockázatkezelésre, a megbízhatóság-elemzésre vonatkozó ismeretek átadását is.

9. RÖVIDÍTÉSEK JEGYZÉKE

EJJT	Elektronikus Jármű- és Járműirányítási Tudásközpont
ETA	Event Tree Analysis – Eseményfa-analízis
FIT	Failures In Time – Meghibásodási gyakoriság mértékegysége
FMEA	Failure Modes And Effects Analysis – Hibamódok és hatások analízise
FTA	Fault Tree Analysis – Hibafa-analízis
MÁV	Magyar Államvasutak
MTBF	Mean Time Between Failures – Két meghibásodás közötti átlagos idő (javítható rendszereknél)
MTTF	Mean Time To Failure – A meghibásodásig eltelt átlagos idő (nem javítható rendszereknél)
MTTR	Mean Time To Repair – A javításig eltelt átlagos idő
RPN	Risk Priority Number
SIL	Safety Integrity Level - Biztonságintegritási szint
THR	Tolerable Hazard Rate - eltűrhető veszélyességi ráta
XOR	Exclusive OR

10. IRODALOMJEGYZÉK

[Apostolakis et. al.,1978]: Apostolakis, G. - S. Garribba - G. Volta, (Eds.): Synthesis and Analysis Methods for Safety and Reliability Studies. *Plenum*, 1978.

[Aven, 1985]: Aven, T.: Reliability evaluation of multistate systems with multistate components. *IEEE Transactions on Reliability*, Vol. R-34, No. 5., pp. 473-479. 1985.

[Bartha et. al., 2005]: Bartha T. – Varga, I. – Soumelidis, A. – **Szabó, G.**: Implementation of a Testing and Diagnostic Concept for an NPP Reactor protection System*. In: Dependable Computing – EDCC-5 (Eds. M. Dal Chin, M. Kaaniche, A. Pataricza). Proceedings of the 5th European Dependable Computing Conference.* Springer, pp. 391-402, Budapest, 2005.

[Bittanti, 1987]: Bittanti, S. (ed.): Software Reliability Modelling and Identification. *Springer-Verlag*, 1987.

[Bokor et. al., 1991]: Bokor, J. - A. Edelmayer - A. Soumelidis - M. Tanyi - P. Gáspár - I. Nagy: Knowledge-Based Noise Analysis: A Promising Tool for Early Failure Detection in Nuclear Power Plants. *Proc. of the IFAC/IMACS Symposium on Fault Detection, SAFEPROCESS'91, Baden-Baden, FRG,* Vol. 2, pp. 73-80. 1991.

[Bokor et. al., 1997]: Bokor, J. - **Szabó G.** - Gáspár P. - Hetthésy J.: Reliability Analysis of Protection Systems in NPPs Using Fault-Tree Analysis Method. *Proceedings of the IAEA Symposium on Computerized Reactor Protection and Safety Related Systems in Nuclear Power Plants*, pp 91-104, Budapest, 1997.

[Brow, 1990]: Brow, K. S.: Evaluating fault trees (and & or gates only) with repeated events. *IEEE Transactions on Reliability*, Vol 39, No.2., pp. 226-235,1990

[Caldarola, 1980]: Caldarola, L.: Coherent systems with multistate components. *Nuclear Engineering and Design*, Vol. 58, pp. 127-139. 1980.

[Chunning és Dinghua, 1990]: Chunning, Y. - S. Dinghua: Classification of fault trees and algorithms of fault tree analysis. *Microelectronics and Reliability*, Vol. 30, No. 5, pp. 891-895. 1990.

[Crosetti és Bruce,1970]: Crosetti, P.A. - R.A. Bruce: Commerical application of fault tree analysis. *Proc. of the Annual Reliability and Maintainability Symp.*, pp. 230-244. 1970.

[Csertán et. al., 1996]: Csertán Gy. – Pataricza A. – Selényi E.: Design for Testability with HW-SW Co-design. *Periodica Polytechnica*, Vol 40(1), pp. 25-37, 1996.

[DIN19250]: DIN V 19250, Grundlegende Sicherheitsbetrachtungen für MSR-Schutzeinrichtungen, 1994.

[Doyle et. al., 1995]: Doyle, S. A. - J. B. Dugan - M. Boyd: Combinatorical models and coverage: a binary decision diagram approach*. Proc. of the Annual Reliability and Maintainability Symposium*, 1995, pp. 82-89.

[Dugan et. al., 1990]: Dugan, J. B. - S. J. Bavuso - M. A. Boyd: Fault trees and sequence dependencies. *Proc. of the Annual Reliability and Maintainability Symp.*, 1990, pp. 286-293.

[EN 50126]: MSZ-EN 50126-1, Vasúti alkalmazások. A megbízhatóság, az üzemkészség, a karbantarthatóság és a biztonság (RAMS) előírása és bizonyítása. 1. rész: Alapvető követelmények és az általános folyamat. 2. kiadás, 2006. november.

[EN 50129]: MSZ-EN 50129, Railway Applications - Communication, Signalling and Processing systems - Safety Related Electronic Systems for Signalling, 2003.

[Garribba et. al., 1985]: Garribba, S. - E. Guagnini - P. Mussio: Multiple-valued logic trees: meaning and prime implicants. *IEEE Transactions on Reliability*, Vol. R-34, No. 5., 1985, pp. 473-472.

[Gáspár és Szabó, 1998a]: Gáspár P. - **Szabó G.**: Analysis of Adaptive Multi-State Logic in Fault-Tolerant Systems. *Proceedings of the Probabilistic Safety Assesment and Management - PSAM 4 Conference*, pp. 13-17, New York, 1998

[Gáspár és Szabó, 1998b]: Gáspár P. - **Szabó G.**: Complex Failure Models for Dependability Assessment. *Digest of FastAbstracts, International Symposium on Fault Tolerant Computing, FTCS-28*, pp 94-95. Munich, 1998.

[Gáspár és Szabó, 1998c]: Gáspár P. - **Szabó G.**: Szoftver alapú rendszerek elvi alapjai - A megbízhatóság és rendelkezésre állás igazolásának lehetõségei és módszerei. *Magyar Tudományos Akadémia Számítástechnikai és Automatizálási Kutató Intézete, MTA SzTAKI, SCL-001/1998.*

[Gáspár és Szabó, 1999a]: Gáspár P. - **Szabó G.**: Automatic Fault-Tree Generation as a Part of a Complex Development System. *Proceedings of the 3rd Inernational Scientific Conference Elektro '99, Section Information & Safety Systems*, pp. 19-24, Zilina, 1999.

[Gáspár és Szabó, 1999b]: Gáspár P. - **Szabó G.**: On-line System Verification Applying an Automatic Fault-Tree Generation Method Integrated into Development Tools. *Proceedings of the European Safety and Reliability Conference-ESREL* pp. 809-814, Münich, 1999.

[Görög et. al., 1998]: Görög B. - **Szabó G.** - Tarnai G.: Biztosítóberendezési funkciók PLC-s megvalósításának biztonsági és megbízhatósági szempontú elemzése. *Vezetékek Világa, Magyar Vasúttechnikai Szemle, 1998. Vol. 3. pp. 6-10.*

[Gulati és Dugan, 1997]: Gulati, R. - J.B. Dugan: A modular approach for analyzing static and dynamic fault trees. *Proc. of the Annual Reliability and Maintainability Symp.*, 1997, pp. 57-63.

[Hartonas-Garmhausen et. al.,1998]: Hartonas-Garmhausen, V – S. Campos – A. Cimatti – E. Clarke – F. Giunchiglia: Verification of a Safety-Critical Railway Interlocking System with Real-Time Constraints. *Digest of Papers, FTCS-28, Munich, Germany*, pp. 458-463. 1998.

[Heger et. al., 1995]: Heger, A.S. - J.K. Bhat - D.W. Stack - D.V. Talbott: Calculating exact top-event probabilities using ΣΠPATREC. *IEEE Transactions on Reliability*, Vol 44, No.4.,1995; pp. 640-644.

[Hudoklin és Rozman, 1985]: Hudoklin, A. - V. Rozman: Safety Analysis of the Railway Traffic System. *Reliability Engineering and System Safety,*. Vol. 37, No. 3., pp. 7-13. 1985.

[Hwang et. al., 1981]: Hwang, C.L. - F.A. Tillman - M.H. Lee: System-reliability evaluation tecniques for complex/large systems – A review. *IEEE Transactions on Reliability*, Vol. R-30, No. 5, 1981, pp. 416-423.

[IEC 61508]: IEC 61508, Functional safety of electrical/electronic/programmable electronic safety related systems. Part 1. - Part 7. 1998, 2003.

[Jain,1997]: Jain, A.: Reliability Prediction. *Annual Reliability and Maintainability Symposium*, 1997.

[Jensen, 1996]: Jensen, H.: The Safety Case - a New European Approach to Guided Transportation Safety Philosophy Demonstrated on Emsland TRANSRAPID and Frankfurt SKY LINE. *Proc. of World Congress on Railway Research*, pp. 173-185. 1996.

[Kai, 1990]: Kai, Yu: Multistate fault-tree analysis*. Reliability Engineering and System Safety*, Vol. 28, 1990, pp. 1-7.

[Kocza és Bossche, 1997]: Kocza G. - A. Bossche: Automatic fault-tree synthesis and real-time tree trimming, based on computer models, *Proc. Ann. Reliability & Maintainability Symp.*, 71-75.,1997

[Lapp és Powers, 1977]: Lapp S. A. - G. J. Powers: Computer aided synthesis of fault trees, *IEEE Transactions on Reliability, April 1977,* 2-13, 1977.

[Lee et. al.,1985]: Lee, D. W. S. - L. Gros - F. A. Tillman - C. H. Lie: Fault Tree Analysis, Methods, and Applications - a Review, *IEEE Transactions on Reliability*, 34:194-203, 1985.

[Leitch, 1995]: Leitch, R. D.: Reliability Analysis for Engineers*. Oxford University Press*, 1995.

[Leveson, 1995]: Leveson, Nancy G.: Safeware. *Addison-Wesley*, 1995

[Liggesmeyer és Rothfelder, 1998]: Liggesmeyer, P. - M. Rothfelder: Improving System Reliability with Automatic Fault Tree Generation. *Proc. of the FTCS-28*, Munich, pp.90-99, 1998.

[MIL-HDBK 217F]: USA Military Handbook (MIL-HDBK) 217F – Reliability Prediction

[MÜ8004]: MÜ8004, Technische Anforderungen an Sicherungsanlagen der Elektronik und Relaistechnik, 1998.

[Patterson-Hine és Koen, 1989]: Patterson-Hine, F.A. - B.V. Koen: Direct evaluation of fault trees using object-oriented programming techniques. *IEEE Transactions on Reliability*, Vol. 38, No. 2., 1989, pp. 186-192.

[Rastocny és Janota, 2000]: Rastocny, K. - A. Janota: Safety Analysis of the Interlocking System. *Transport (Politechnica Warszawa, ISSN 1230-9265),* Vol 44. pp. 89-104, 2000.

[Relcon]: Home Page of the Relcon Company, Sweden. http://www.relcon.se

[Renault et. al.,1999]: Renault I. - M. Pilliere - N. Villatte - P. Mouttapa: KB3: Computer program for automatic generation of fault trees. *Proc. Ann. Reliability & Maintainability Symp.*, 389-395, 1999.

[RiskSpectrum a]: RiskSpectrum Users Manual. Relcon, Sweden.

[RiskSpectrum b]: RiskSpectrum Theory Manual. Relcon, Sweden

[RPP Bellcore]: Reliability Prediction Procedure (RPP), Bellcore Inc., TR-332, Issue 5.

[Schaefer, 1983]: Schaefer, E.: Megbízhatóság az elektronikában. *Műszaki Könyvkiadó*, 1983

[Schneeweis, 1985]: Schneeweis, W. G.: Fault-tree analysis using a binary decision tree. *IEEE Transactions on Reliability*, Vol. R-34, No. 5., 1985, pp. 453-457.

[Schneeweis, 1987]: Schneeweis, W. G.: Approximate fault-tree analysis with prescribed accuracy. *IEEE Transactions on Reliability*, Vol. R-36, No. 2., 1987, pp. 250-254.

[Soumelidis et. al., 1994]: Soumelidis, A. - J. Bokor - L. Keviczky - A. Edelmayer - P. Gáspár - Zs. Csáki - E. Varga: Toward An Intelligent Evolutionary Signal Processing Based Failure Monitoring and Diagnostic System for Complex Plants. *Proc. of the IFAC/IMACS Symposium on Fault Detection, SAFEPROCESS'94,* pp. 760-765, 1994.

[SRP Bellcore]: Generic Requirements for Software Reliability Prediction (SRP), Bellcore Inc., GR-2813-CORE

[Stecher, 1986]: Stecher, K.: Evaluation of large fault-trees with repeated events using an efficient bottom-up algorithm. *IEEE Transactions on Reliability*, Vol. R-35, No. 1., 1986, pp. 51-58.

[Storey, 1996]: Storey, N: Safety-Critical Computer Systems. *Addison-Wesley*, 1996.

[Szabó és Csiszár, 2000a]: **Szabó G.** – Csiszár Z.: Fault-Tree Synthesis: a Practical Approach. *TU Budapest, Research News, Special Issue 2000.*

[Szabó és Csiszár, 2000b]: **Szabó G.** – Csiszár Z.: Automatikus hibafa generálás – Tanszéki kutatási jelentés. *BME Közlekedésautomatikai Tanszék*, 2000.

[Szabó és Gáspár, 1998a]: **Szabó G.** - Gáspár P.: Probabilistic Dependability Analysis of Adaptive Functions: A Fault-Tree Based Approach and Its Application in Transportation. *Periodica Politachnica Ser. Transp. Eng., 1998. Vol. 26, No 1-2, pp. 187-200.*

[Szabó és Gáspár, 1998b]: **Szabó G.** - Gáspár P.: Practical Aspects of Dependability Analysis for Vehicle Systems. *Proceedings of the 6th Mini Conference on Vehicle System Dynamics, Identification, and Anomalies, VSDIA*, pp. 437-446. Budapest, 1998.

[Szabó és Gáspár, 1999a]: **Szabó G.** - Gáspár P.: Fault-tree analysis of System Functionality modelled as Binary Adaptive Functions*. Proceedings of the European Safety and Reliability Conference-ESREL* pp. 1033-1038, Münich, 1999.

[Szabó és Gáspár, 1999b]: **Szabó G.** - Gáspár P.: Practical Treatment-Methods of Adaptive Components in the Fault-Tree Analysis*. Proceedings of the Annual Reliability and Maintainability Symposium*, pp. 97-104, Washington D.C., 1999

[Szabó et. al., 2008]: **Szabó G.** – Sághi B. –Darai L.– Jakubovics J. – Héray T. – Kirilly K. – Buzás M. – Gál I.: Biztosítóberendezések időszakos vizsgálatainak koncepciója. *Vezetékek Világa, Magyar Vasúttechnikai Szemle*, 2008.

[Szabó et. al., 2004]: **Szabó G.** – Szabó K. – Zerényi R.: Safety Management Systems in Transportation: Aims and Solutions. *Periodica Politechnica, Ser. Transp. Eng, 2004. Vol. 32. No. 1-2,* pp. 123-134., 2004.

[Szabó és Tarnai, 1999]: **Szabó G.** - Tarnai G.: Dependability Analysis of Interlocking Systems - A Comparison of the Probabilistic and the Deterministic Approaches. *Proceedings of the 3rd Inernational Scientific Conference Elektro '99, Section Information & Safety Systems*, pp. 7-12, Zilina, 1999.

[Szabó és Tarnai, 2000]: **Szabó G.** - Tarnai G.: Automatic Fault-Tree Generation as a Support for Safety Studies of Railway Interlocking Systems*. Proceedings of the IFAC Symposium on Control in Transportation Systems*, pp. 453-458, Braunschweig, 2000.

[Szabó és Tarnai, 2002]: **Szabó G.** – Tarnai G.: A vasúti biztosítóberendezések biztonságigazolási módszereinek fejlődése, az új, eurokonform szabályozás alkalmazásának kérdései. *Vezetékek Világa, Magyar Vasúttechnikai Szemle, 2002/4. szám*, 5-9 oldal, 2002.

[Szabó és Tarnai, 2003]: **Szabó G.** – Tarnai G.: A vasúti biztonság bizonyítására vonatkozó új európai szabványok alkalmazási kérdései. *Vezetékek Világa, Magyar Vasúttechnikai Szemle, 2003/1. szám*, 2-6 oldal, 2003.

[Szabó, 1995]: **Szabó G**.: Bevezetés a hibafa-analízisbe. Oktatási segédlet*. BME Közlekedésautomatikai Tanszék*, 1996.

[Szabó, 2007a]: **Szabó G**.: Kockázati alapú fejlesztési kritériumok a járművek biztonsági rendszereinél. *Jövő Járműve, 2007/1-2 szám*, 38-41 oldal, 2007.

[Szabó, 2007b]: **Szabó G.**: Műszaki okú kockázatok kezelése a közlekedésben. *Innováció és fenntartható felszíni közlekedés c. konferencia. Magyar Mérnöki Akadémia, 2008.* Az előadás anyaga elektronikusan elérhető: http://kitt.bmf.hu/mmaws/2007/pages/participants.html (az elektronikus elérhetőség utolsó validálása 2007-ben).

[Szabó, 2008]: **Szabó, G.**: Setting Up the Concept of Periodic Testing and Examinations of Safety Systems. *In: Formal Methods for Automation and Safety in Railway and Automotive Systems (Eds. G. Tarnai, E. Schnieder). Proceedings of Symposium FORMS/FORMAT2008.* pp. 321-324, Budapest, 2008.

[Tarnai a]: Tarnai G.: Harmonisation Method of Validation of Safety Systems *Komunikacie/Communications - Scientific Letters of the University of Zilina, Slovakia.* http://www1.kka.bme.hu/~tarnai/papers/paper009.htm, (az elektronikus elérhetőség utolsó validálása 2008-ban).

[Tarnai b]: Tarnai G.: A 75 Hz-es táplálás biztonsági követelményosztályának meghatározása. *Elektronikus közlemény.* http://www1.kka.bme.hu/~tarnai/papers/paper008.htm, (az elektronikus elérhetőség utolsó validálása 2008-ban).

[Tarnai c]: Tarnai G.: Biztonságigazolási rendszerek harmonizálása. *Elektronikus közlemény.* http://www1.kka.bme.hu/~tarnai/papers/paper006.htm, (az elektronikus elérhetőség utolsó validálása 2008-ban).

[Tarnai d]: Tarnai G.: Közlekedési automatika. *Elektronikus közlemény.* http://www1.kka.bme.hu/~tarnai/papers/paper024.htm, (az elektronikus elérhetőség utolsó validálása 2008-ban).

[Tobias és Trindade, 1998]: Tobias, P. –D. Trindade: Applied Reliability. *Chapman &Hall /CRC*, 1998.

[Verumi et. al.,1999]: Verumi, K. K. - J. B. Dugan - K. J. Sullivan: A design language for automatic synthesis of fault trees, *Proc. Ann. Reliability & Maintainability Symp.*, 91-96, 1999.

[Wood, 1985]: Wood, A. P.:Multistate block diagrams and fault trees*. IEEE Transactions on Reliability*, Vol. R-34, No. 3., 1985, pp. 236-240.

11. MELLÉKLETEK

1. melléklet: Zárt alakú képletek három bemenet esetén

VAGY hibafa-kapu 3 bemenettel

$$\begin{aligned}P_{OR}^{Normal} &= P_{D_1} + P_{D_2} + P_{D_3} + P_{N_1} + P_{N_2} + P_{N_3} - P_{N_1}P_{N_2} - P_{N_1}P_{N_3} - P_{N_1}P_{D_2} - P_{N_1}P_{D_3} - P_{D_1}P_{N_2} - \\ &\quad P_{D_1}P_{N_3} - P_{D_1}P_{D_2} - P_{D_1}P_{D_3} - P_{N_2}P_{N_3} - P_{N_2}P_{D_3} - P_{D_2}P_{N_3} - P_{D_2}P_{D_3} + P_{N_1}P_{N_2}P_{N_3} + \\ &\quad P_{D_1}P_{D_2}P_{N_3} + P_{D_1}P_{N_2}P_{N_3} + P_{D_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{N_3} + P_{N_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{D_3} + P_{D_1}P_{D_2}P_{D_3} \\ &= \sum_{\substack{P_{A\ell}\in\{P_{D\ell},P_{N\ell}\} \\ \ell\in\{1\ldots3\}}} P_{A\ell} - \sum_{\substack{P_{A\ell_1},P_{A\ell_2}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2}} P_{A\ell_1}P_{A\ell_2} + \sum_{\substack{P_{A\ell_1},P_{A\ell_2},P_{A\ell_3}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2,\ell_3\in\{1\ldots3\},\ell_1<\ell_2<\ell_3}} P_{A\ell_1}P_{A\ell_2}P_{A\ell_3} \\ &= \sum_{m=1}^{3}\left[(-1)^{m+1}\sum_{\substack{P_{A\ell_j}\in\{P_{D\ell_j},P_{N\ell_j}\} \\ \ell_j\in\{1\ldots3\},\ell_1<\ldots<\ell_m}}\prod_{j=1}^{m}(P_{A\ell_j})\right]\end{aligned}$$

(M1-1.)

$$\begin{aligned}P_{Det,OR}^{Active} &= P_{D_1} + P_{D_2} + P_{D_3} - P_{D_1}P_{D_2} - P_{D_1}P_{D_3} - P_{D_2}P_{D_3} + P_{D_1}P_{D_2}P_{D_3} \\ &= \sum_{\ell=1}^{3}P_{D\ell} - \sum_{\ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2}P_{D\ell_1}P_{D\ell_2} + \prod_{j=1}^{3}P_{Dj} = \sum_{m=1}^{3}\left[(-1)^{m+1}\sum_{\ell_j\in\{1\ldots3\},\ell_1<\ldots<\ell_m}\prod_{j=1}^{m}(P_{D\ell_j})\right]\end{aligned}$$

(M1-2.)

$$P_{Det,OR}^{Adaptive} = P_{D_1}P_{D_2}P_{D_3} = \prod_{j=1}^{3}(P_{Dj})$$ **(M1-3.)**

$$\begin{aligned}P_{UnDet,OR}^{Adaptive} &= P_{N_1} + P_{N_2} + P_{N_3} - P_{N_1}P_{N_2} - P_{N_1}P_{N_3} - P_{N_2}P_{N_3} + P_{N_1}P_{N_2}P_{N_3} \\ &= \sum_{\ell=1}^{3}P_{N\ell} - \sum_{\ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2}P_{N\ell_1}P_{N\ell_2} + \prod_{j=1}^{3}P_{Nj} = \sum_{m=1}^{3}\left[(-1)^{m+1}\sum_{\ell_j\in\{1\ldots3\},\ell_1<\ldots<\ell_m}\prod_{j=1}^{m}(P_{N\ell_j})\right]\end{aligned}$$

(M1-4.)

$$P_{\text{UnDet,OR}}^{\text{Active}} = P_{N_1} + P_{N_2} + P_{N_3} - P_{N_1}P_{N_2} - P_{N_1}P_{N_3} - P_{N_1}P_{D_2} - P_{N_1}P_{D_3} - P_{D_1}P_{N_2} - P_{D_1}P_{N_3} - P_{N_2}P_{N_3} - P_{N_2}P_{D_3} - P_{D_2}P_{N_3} + P_{N_1}P_{N_2}P_{N_3} + P_{D_1}P_{D_2}P_{N_3} + P_{D_1}P_{N_2}P_{N_3} + P_{D_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{N_3} + P_{N_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{D_3}$$

$$= \sum_{\ell=1}^{3} P_{N\ell} - \sum_{\substack{P_{A\ell_1},P_{A\ell_2}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2 \\ \exists\ell_s\in\{\ell_1,\ell_2\}:\,P_{A\ell_s}=P_{N\ell_s}}} P_{A\ell_1}P_{A\ell_2} + \sum_{\substack{P_{A\ell_1},P_{A\ell_2},P_{A\ell_3}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2,\ell_3\in\{1\ldots3\},\ell_1<\ell_2<\ell_3}} P_{A\ell_1}P_{A\ell_2}P_{A\ell_3}$$

$$= \sum_{m=1}^{3}\left[(-1)^{m+1}\sum_{\substack{P_{A\ell_j}\in\{P_{N\ell_j},P_{D\ell_j}\} \\ \ell_j\in\{1\ldots3\},\ell_1<\ldots<\ell_m \\ \exists\ell_s\in\{\ell_1\ldots\ell_m\}:\,P_{A\ell_s}=P_{N\ell_s}}}\prod_{j=1}^{m}(P_{A\ell_j})\right]$$

(M1-5.)

ÉS hibafa-kapu három bemenettel

$$P_{\text{AND}}^{\text{Normal}} = P_{N_1}P_{N_2}P_{N_3} + P_{D_1}P_{D_2}P_{N_3} + P_{D_1}P_{N_2}P_{N_3} + P_{D_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{N_3} + P_{N_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{D_3} + P_{D_1}P_{D_2}P_{D_3} = \sum_{\substack{P_{A\ell_1},P_{A\ell_2},P_{A\ell_3}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2,\ell_3\in\{1\ldots3\},\ell_1<\ell_2<\ell_3}} P_{A\ell_1}P_{A\ell_2}P_{A\ell_3} = \sum_{P_{Aj}\in\{P_{Nj},P_{Dj}\}}\prod_{j=1}^{3}(P_{Aj})$$

(M1-6.)

$$P_{\text{Det,AND}}^{\text{Active}} = P_{D_1}P_{D_2}P_{D_3} = \prod_{j=1}^{3}(P_{Dj}) = P_{\text{Det,AND}}^{\text{Adaptive}}$$

(M1-7.)

$$P_{\text{Un det,AND}}^{\text{Active}} = P_{N_1}P_{N_2}P_{N_3} + P_{D_1}P_{D_2}P_{N_3} + P_{D_1}P_{N_2}P_{N_3} + P_{D_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{N_3} + P_{N_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{D_3} = \sum_{\substack{P_{A\ell_1},P_{A\ell_2},P_{A\ell_3}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2,\ell_3\in\{1\ldots3\},\ell_1<\ell_2<\ell_3 \\ \exists j:\,P_{Aj}=P_{Nj}}} P_{A\ell_1}P_{A\ell_2}P_{A\ell_3} = \sum_{\substack{P_{Aj}\in\{P_{Nj},P_{Dj}\} \\ \exists j:\,P_{Aj}=P_{Nj}}}\prod_{j=1}^{3}(P_{Aj}) = P_{\text{Un det,AND}}^{\text{Adaptive}}$$

(M1-8.)

2v3 hibafa-kapu

$$P_{2v3}^{Normal} = P_{N_1}P_{N_2} + P_{N_1}P_{N_3} + P_{N_1}P_{D_2} + P_{N_1}P_{D_3} + P_{D_1}P_{N_2} + P_{D_1}P_{N_3} + P_{N_2}P_{N_3} + P_{N_2}P_{D_3} + P_{D_2}P_{N_3} + P_{D_1}P_{D_2} + P_{D_1}P_{D_3} + P_{D_2}P_{D_3} - 2P_{N_1}P_{N_2}P_{N_3} - 2P_{D_1}P_{D_2}P_{N_3} - 2P_{D_1}P_{N_2}P_{N_3} - 2P_{D_1}P_{N_2}P_{D_3} - 2P_{N_1}P_{D_2}P_{N_3} - 2P_{N_1}P_{N_2}P_{D_3} - 2P_{N_1}P_{D_2}P_{D_3} - 2P_{D_1}P_{D_2}P_{D_3}$$

$$= \sum_{\substack{P_{A\ell_1},P_{A\ell_2}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2}} P_{A\ell_1}P_{A\ell_2} - \sum_{\substack{P_{A\ell_1},P_{A\ell_2},P_{A\ell_3}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2,\ell_3\in\{1\ldots3\},\ell_1<\ell_2<\ell_3}} 2P_{A\ell_1}P_{A\ell_2}P_{A\ell_3}$$

$$= \sum_{m=2}^{3}\left[(k-1)(-1)^m \sum_{\substack{P_{A\ell_j}\in\{P_{D\ell_j},P_{N\ell_j}\} \\ \ell_j\in\{1\ldots3\},\ell_1<\ldots<\ell_m}} \prod_{j=1}^{m}(P_{A\ell_j})\right]$$

(M1-9.)

$$P_{Det,2v3}^{Active} = P_{D_1}P_{D_2} + P_{D_1}P_{D_3} + P_{D_2}P_{D_3} - 2P_{D_1}P_{D_2}P_{D_3}$$

$$= \sum_{\ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2} P_{D\ell_1}P_{D\ell_2} - 2P_{D_1}P_{D_2}P_{D_3} = \sum_{m=2}^{3}\left[(-1)^m(k-1) \sum_{\ell_j\in\{1\ldots3\},\ell_1<\ldots<\ell_m} \prod_{j=1}^{m}(P_{D\ell_j})\right]$$

(M1-10.)

$$P_{UnDet,2v3}^{Active} = P_{N_1}P_{D_2} + P_{N_1}P_{D_3} + P_{D_1}P_{N_2} + P_{D_1}P_{N_3} + P_{N_2}P_{D_3} + P_{D_2}P_{N_3} + P_{N_1}P_{N_2} + P_{N_1}P_{N_3} + P_{N_2}P_{N_3} - 2P_{N_1}P_{N_2}P_{N_3} - 2P_{D_1}P_{D_2}P_{N_3} - 2P_{D_1}P_{N_2}P_{N_3} - 2P_{D_1}P_{N_2}P_{D_3} - 2P_{N_1}P_{D_2}P_{N_3} - 2P_{N_1}P_{N_2}P_{D_3} - 2P_{N_1}P_{D_2}P_{D_3} = \sum_{\substack{P_{A\ell_1},P_{A\ell_2}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2}} P_{A\ell_1}P_{A\ell_2} - \sum_{\substack{P_{A\ell_1},P_{A\ell_2},P_{A\ell_3}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2,\ell_3\in\{1\ldots3\},\ell_1<\ell_2<\ell_3 \\ \exists\ell_s\in\{\ell_1,\ell_2,\ell_3\}:P_{A\ell_s}=P_{N\ell_s}}} 2P_{A\ell_1}P_{A\ell_2}P_{A\ell_3}$$

$$= \sum_{m=2}^{3}\left[(-1)^m(m-1) \sum_{\substack{P_{A\ell_j}\in\{P_{N\ell_j},P_{D\ell_j}\} \\ \ell_j\in\{1\ldots3\}:\ell_1<\ldots<\ell_m \\ \exists\ell_s\in\{\ell_1\ldots\ell_m\}:P_{A\ell_s}=P_{N\ell_s}}} \prod_{j=1}^{k}(P_{A\ell_j})\right]$$

(M1-11.)

$$P_{Det,2v3}^{Adaptive} = P_{D_1}P_{D_2}P_{D_3} = \prod_{j=1}^{3}(P_{Dj})$$

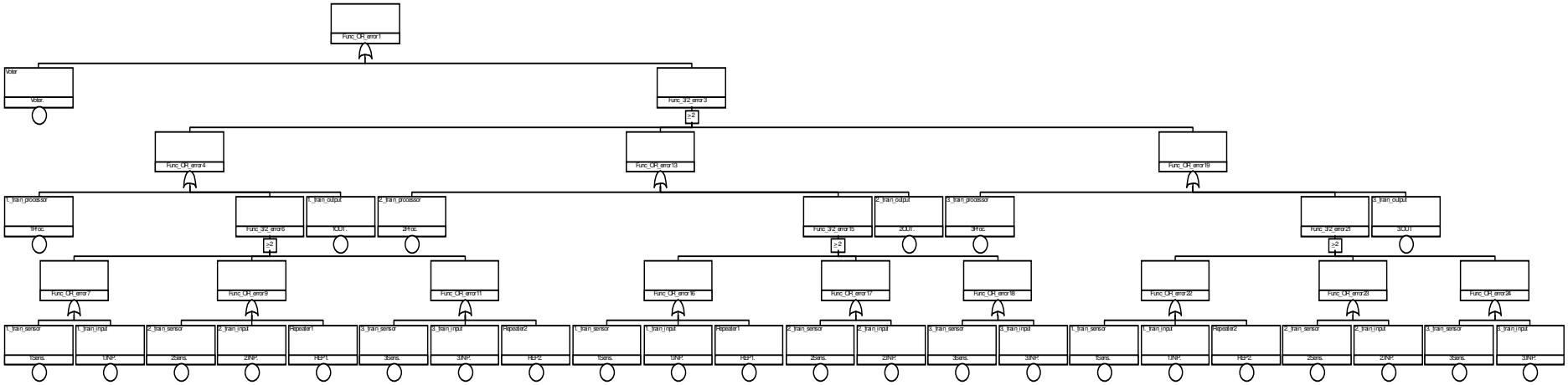
(M1-12.)

$$
\begin{aligned}
P_{\mathrm{UnDet},2v3}^{\mathrm{Adaptive}} &= P_{N_1}P_{N_2} + P_{N_1}P_{N_3} + P_{N_2}P_{N_3} + P_{D_1}P_{D_2}P_{N_3} + P_{D_1}P_{N_2}P_{D_3} + P_{N_1}P_{D_2}P_{D_3} - 2P_{N_1}P_{N_2}P_{N_3} \\
&= \sum_{\ell_1,\ell_2\in\{1\ldots3\},\ell_1<\ell_2} P_{N\ell_1}P_{N\ell_2} + \sum_{\substack{P_{A\ell_1},P_{A\ell_2},P_{A\ell_3}\in\{P_{D\ell},P_{N\ell}\} \\ \ell_1,\ell_2,\ell_3\in\{1\ldots3\},\ell_1<\ell_2<\ell_3 \\ \exists\ell_s\in\{\ell_1,\ell_2,\ell_3\}:P_{A\ell_s}=P_{N\ell_s} \\ \forall\ell_z\in\{\ell_1,\ell_2,\ell_3\}:\ell_z\neq\ell_s:P_{A\ell_z}=P_{D\ell_z}}} P_{A\ell_1}P_{A\ell_2}P_{A\ell_3} - 2P_{N_1}P_{N_2}P_{N_3} \\
&= \sum_{m=2}^{3}\left[(m-1)(-1)^m \sum_{\ell_j\in\{1\ldots3\}:\ell_1<\ldots<\ell_m} \prod_{j=1}^{m}(P_{N\ell_j})\right] + \sum_{\substack{P_{A\ell_j}\in\{P_{N\ell_j},P_{D\ell_j}\} \\ \ell_j\in\{1\ldots3\}:\ell_1<\ldots<\ell_3 \\ \exists\ell_s\in\{\ell_1\ldots\ell_3\}:P_{A\ell_s}=P_{N\ell_s} \\ \forall\ell_z\in\{\ell_1\ldots\ell_3\}:\ell_z\neq\ell_s:P_{A\ell_z}=P_{D\ell_z}}} \prod_{j=1}^{3}(P_{A\ell_j})
\end{aligned}
$$

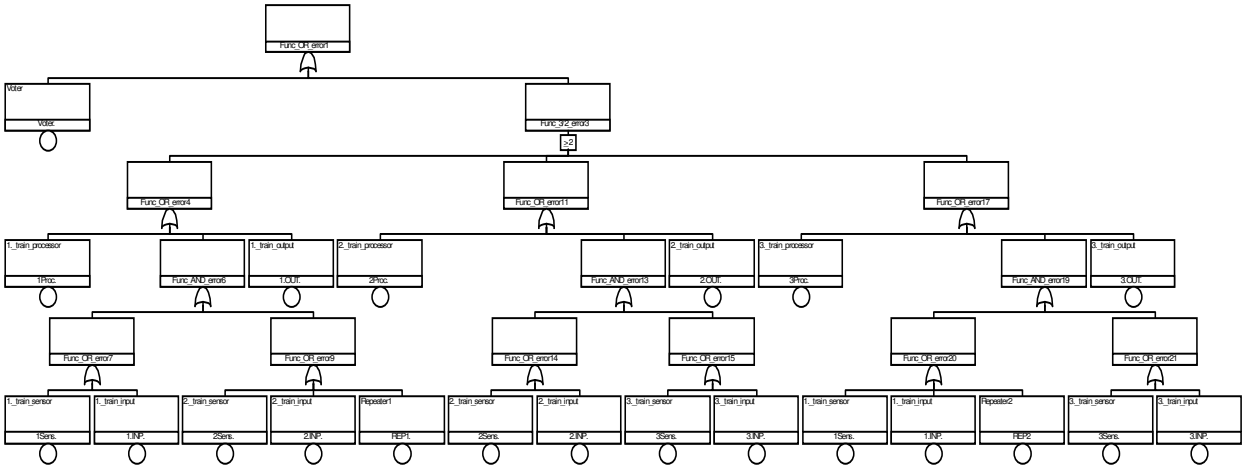
(M1-13.)

2. melléklet: Demonstrációs példa az automatikus hibafa-generálásra. Szabálygyűjtemény és hibafák

```
//
// Rule-collection, Functional and HW
description
//
Functional description
501. INP 101 0
601. INP 201 0
701. INP 301 0
502. 2/N 103 3 501 601 701
602. 2/N 203 3 501 601 701
702. 2/N 303 3 501 601 701
800. 2/N 1000 3 502 602 702
801. OUT 1000 1 800
//
HW description
101. PROBABILITY 1Sens. 1._train_sensor
201. PROBABILITY 2Sens. 2._train_sensor
301. PROBABILITY 3Sens. 3._train_sensor
102. PROBABILITY 1Inp.  1._train_input
202. PROBABILITY 2Inp.  2._train_input
302. PROBABILITY 3Inp.  3._train_input
103. PROBABILITY 1Proc. 1._train_processor
203. PROBABILITY 2Proc. 2._train_processor
303. PROBABILITY 3Proc. 3._train_processor
104. PROBABILITY 1Out.  1._train_output
204. PROBABILITY 2Out.  2._train_output
304. PROBABILITY 3Out.  3._train_output
110. PROBABILITY Rep1.  Repeater1
111. PROBABILITY Rep2.  Repeater2
1000. PROBABILITY Voter.  Voter2
//
Rule base // Actuation masking
GATE AND OR Func_AND_error
GATE OR AND Func_OR_error
GATE 2/N 2/N Func_3/2_error
//
Parameters
1. Sensor_P PROBABILITY 0.02
2. Input_P PROBABILITY 0.01
3. Computer_P PROBABILITY 0.07
4. Output_P PROBABILITY 0.05
5. Non_used_P PROBABILITY 0.00
//
HW connections
101. 102
102. 101 103
103. 102 104 110 111
104. 103 1000
110. 103 203
111. 103 303
201. 202
202. 201 203
203. 202 204 110
204. 203 1000
301. 302
302. 301 303
303. 302 304 111
304. 303 1000
1000. 104 204 304
//
HW parameters
101. 1
102. 2
103. 3
104. 4
110. 3
111. 3
201. 1
202. 2
203. 3
204. 4
301. 1
302. 2
303. 3
304. 4
1000. 3
```



M2-1. ábra: 3-as redundáns rendszer hibafa-modellje (teljes funkcionalitás)



M2-2. ábra: 3-as redundáns rendszer hibafa-modellje (csökkentett funkcionalitás)

3. melléklet: Veszélyeztetettség függvény monoton növekvő voltának bizonyítása

A deriválás elvégzéséhez először írjuk fel a függvényt részfüggvények szorzataként. A részfüggvényre bontás alapja az $n \to \infty$ vizsgálatnál alkalmazott részhatárérték-felbontás:

$$P_{\max} = x(n) \cdot y(n) \cdot z(n) \qquad \textbf{(M3-1.)}$$

$$\frac{P_{felső}}{dn} = \frac{d(x(n))}{dn} \cdot y(n) \cdot z(n) + x(n) \cdot \frac{d(y(n))}{dn} \cdot z(n) + x(n) \cdot y(n) \cdot \frac{d(z(n))}{dn} \qquad \textbf{(M3-2.)}$$
$$= a(n) + b(n) + c(n)$$

$$\frac{d(x(n))}{dn} = \frac{d\left(\frac{n}{1000 \cdot n + 1}\right)}{dn} = \frac{1}{(1000 \cdot n + 1)^2} \qquad \textbf{(M3-3.)}$$

$$\frac{d(y(n))}{dn} = \frac{d\left(\left(1 - \frac{1}{1000 \cdot n + 1}\right)^{(n-1)}\right)}{dn} = \frac{d\left(e^{(n-1)\cdot \ln\left(1 - \frac{1}{1000 \cdot n + 1}\right)}\right)}{dn} = \qquad \textbf{(M3-4.)}$$
$$= \left(1 - \frac{1}{1000 \cdot n + 1}\right)^{(n-1)} \cdot \left(\ln\left(1 - \frac{1}{1000 \cdot n + 1}\right) + \frac{1000 \cdot (n-1)}{(1000n+1)^2 - (1000n+1)}\right)$$

$$\frac{d(z(n))}{dn} = \frac{d\left(1 - e^{-\frac{n-1}{1000 \cdot n}}\right)}{dn} = -e^{-\frac{n-1}{1000 \cdot n}} \cdot \left(-\frac{1000 \cdot n - 1000 \cdot (n-1)}{(1000 \cdot n)^2}\right) = \frac{1}{1000 \cdot n^2} \cdot e^{-\frac{n-1}{1000 \cdot n}} \qquad \textbf{(M3-5.)}$$

és a rész-összegek:

$$a(n) = \frac{1}{(1000 \cdot n + 1)^2} \cdot \left(1 - \frac{1}{1000 \cdot n + 1}\right)^{(n-1)} \cdot \left(1 - e^{-\frac{n-1}{1000 \cdot n}}\right) \qquad \textbf{(M3-6.)}$$

$$b(n) = \frac{n}{1000 \cdot n + 1} \cdot \frac{d(y(n))}{dn} \cdot \left(1 - e^{-\frac{n-1}{1000 \cdot n}}\right) \qquad \textbf{(M3-7.)}$$

$$c(n) = \frac{n}{1000 \cdot n + 1} \cdot \left(1 - \frac{1}{1000 \cdot n + 1}\right)^{(n-1)} \cdot \frac{dz(n)}{dn} \qquad \textbf{(M3-8.)}$$

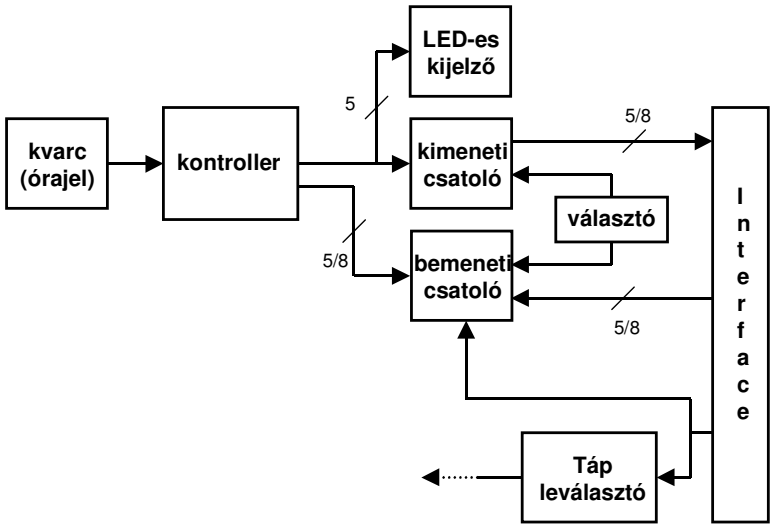
Mivel a derivált értéke n=2-től (a soros rendszer miatt *n* minimális értéke 2), pozitív így $\left(P_{felső}\right)' = 0$ csak $n \to \infty$ esetén teljesül, vagyis az elemzésünkben n minden határon túl való növelésével a maximális biztonság hiány szintet határoztuk meg.

4. melléklet: Példa megbízhatóság becslésére

A példa bevezetése

Az alábbi példában a disszertációban bemutatott elméleti módszerek gyakorlati alkalmazásának demonstrálása céljából egy vasúti automatikai berendezés (75Hz-es biztonsági ütemadó) egyik alrendszerének (jelgeneráló modul) megbízhatósági becslését adjuk meg. A példa valós ipari feladat megoldásán alapul: A berendezést a PowerQuattro Zrt. tervezte és gyártja a MÁV Zrt. részére. A tervezési munkában jelen disszertáció szerzője mint független elemző és validáló vett részt [4M-1].

A modul rendelkezésre állásának becslését két esetre végezzük el a MIL-HDBK 217F alkatrész számbavétel módszere alapján: Az első esetben csak a fő funkcionális elemek meghibásodási rátáit összegezzük, míg a második esetben az összes, kártyán található alkatrész meghibásodását figyelembe vesszük. Mindkét esetben kihasználtuk azt a tényt, hogy a beépített alkatrészeket a tervező a határértékek alatt üzemelteti – így a kiválasztott megbízhatóság-becslési módszer alkalmazható. A jelgeneráló modul funkcionálisan a 4M-1. ábra szerint modellezhető [4M-2].



4M-1. ábra: a jelgeneráló modul funkcionális modellje

A modulban találhatóak a működéshez elengedhetetlenül fontos funkcionális blokkok:

- Órajel-generátor
- Mikrokontroller,
- Be/Kimeneti csatolók és választó,
- Galvanikus elválasztást biztosító egység,
- Csatlakozó (interface),

A megbízhatósági becslésnél a három szárazföldi kategória közül (4M-1. táblázat) a rögzített alkalmazások csoportja tűnik a leírások szerint legalkalmasabbnak a vizsgálat elvégzéséhez, mivel az enyhe igénybevételű kategória felsorolás szerint nem tartalmaz ipari alkalmazási csoportokat. A rögzített alkalmazások becslésénél, a

diszkrét elemek esetén a nagy integráltságú eszközök esetén 60°C-os környezeti hőmérsékletet feltételeznek. A 40°C-os hőmérséklet megfelelhet a jelfogótermi elhelyezés szélső értékének, míg a nagy integráltságú eszközök esetén a saját disszipált teljesítmény környezeti (még inkább: lapkahőmérséklet) emelő hatásának figyelembevétele magyarázza a magasabb hőfokot.

4M-1. táblázat: MIL-HDBK 217F szárazföldi alkalmazási csoportjai

1.	Szárazföldi, enyhe igénybevételű	G_B G_{MS}	Nem mobil, hőmérséklet- és páratartalom szabályozott környezet. Karbantarthatósági szempontból könnyű hozzáférhetőség. Példák: laboratóriumi és tesztberendezések, orvosi berendezések, üzleti és tudományos számítógépek.
2.	Szárazföldi, rögzített	G_F	Mérsékelten szabályozott környezet, pl. állandó helyű, megfelelő hűtésű ipari szekrényben történő felszerelés, pl. légiirányítási radarberendezés vagy távközlési berendezések.
3.	Szárazföldi, mozgó	G_M M_P	Önjáró vagy vontatott járműre installált berendezések, kézzel szállított berendezések, pl. mobil kommunikációs eszközök, lézerkeresők stb.

Mivel a tervekből, illetve a szerelt panelek szemrevételezésével az alkalmazott alkatrészek minősége nem állapítható meg, általános ipari minőséget feltételezünk, amit a becslésnél a legmagasabb minőségi faktor módosító tényezőkkel veszünk figyelembe.

Szükséges megjegyezni, hogy a környezeti hőmérséklet csökkentésével a berendezés megbízhatósága javul. Így egy 25 °C-os hőmérsékletű, légkondicionált helységben való elhelyezés 2-3 szoros megbízhatósági javulást eredményezhet.

Modellezési megfontolások

A megbízhatósági becslés szempontjából két kritikus elem található az alkalmazott alkatrészek között:

- Mikrokontroller, és
- DC/DC átalakító.

Mindkettő olyan összetett elem, amelynek a meghibásodási rátája közvetlenül nem állapítható meg.

A mikrokontroller olyan eszköz, amely minden olyan egységet tartalmaz, amely egy processzoros alkalmazás felépítéséhez szükséges.
Fő egységei:

- 8 bites mikroprocesszor,
- 4kByte Flash EEPROM,
- 256 byte RAM,
- 2x8 bit külső kommunikációs vonal,
- megszakítás-vezérlő,

- időzítők.

A megbízhatósági modellezéshez az alábbi három egység kapcsolataként értelmezzük a mikrokontrollert:
- 8 bites mikroprocesszor,
- 4kByte Flash EEPROM,
- 256 byte RAM,

mivel az összes többi egység egy átlagos processzorban előfordulhat.

A DC/DC átalakítót nem szükséges modellezni, mivel a gyártó cég MIL-HDBK 217E alapú megbízhatósági becslést ad az eszközre:

MTBF $>1.1 \times 10^6$ óra (25^oC-on).

A TIMER modul csatlakozóját nem vesszük figyelembe a megbízhatósági becslésnél, mivel a 40 pólusú csatlakozó csatlakozópontjai páronként kerülnek felhasználásra, és így két csatlakozópont használhatatlansága esetén szűnik csak meg a kapcsolat.

Modul megbízhatóság

A 4M-2. táblázat tartalmazza a TIMER modulban alkalmazott, a funkcionális működést befolyásoló alkatrészek listáját, míg a 4M-3. táblázat az összes modulba épített alkatrész listáját. Mindkét táblázatban szerepelnek a becslési értékek is.

A táblázat jelölései:
- Típus: az adott alkatrész típusa,
- Darabszám: típuson belüli darabszám,
- Hivatkozás: A MIL-HDBK-217F megfelelő szekciója,
- λg: A típushoz tartozó általános meghibásodási ráta (10^6 óra alatti meghibásodások száma /10^6 óra ≈ 41666 nap ≈ 114 év/),
- Q: A típushoz tartozó minőségi tényező,
- Összeg: az adott típusra vonatkozó eredő meghibásodási ráta, amely az általános meghibásodási ráta, a minőségi tényező és a darabszám szorzata.

4M-2. táblázat: TIMER modul megbízhatósági becslése (funkcionálisan érintett részek), szárazföldi, rögzített felhasználási csoport esetén

	Típus	*Darab*	*Hivatkozás*	*λg*	*Q*	*Összeg*
	Mikrokontroller (8 bites mikroprocesszor)	1	5.1	0.089	10	0.89
	Mikrokontroller (4kbyte FLASH)	1	5.2	0.018	10	0.18
	Mikrokontroller (128 byte RAM)	1	5.2	0.022	10	0.22
	Optocsatoló	9	6.11	0.07	8	4.41
	DC/DC konverter	1	0.9			0.90
	Rezgőkvarc	1	19.1	0.096	2.1	0.20
	Egyedi ellenállás	13	9.1	0.016	10	2.08
	Ellenállás létrában	5	9.1	0.016	10	0.75
	Választókapcsoló	3	17.1	0.12	1	0.36

	Típus	*Darab*	*Hivatkozás*	*λg*	*Q*	*Összeg*
	Kondenzátor	6	10.1	0.0064	10	0.38
	Összesen:					10.37
		$10.37*10^{-6}$ 1/óra				
		MTBF=$9.64* 10^{4}$ óra				

4M-3. táblázat: TIMER modul megbízhatósági becslése (összes komponens), szárazföldi, rögzített felhasználási csoport esetén

	Típus	*Darab*	*Hivatkozás*	*λg*	*Q*	*Összeg*
	Mikrokontroller (8 bites mikroprocesszor)	1	5.1	0.089	10	0.89
	Mikrokontroller (4kbyte FLASH)	1	5.2	0.018	10	0.18
	Mikrokontroller (128 byte RAM)	1	5.2	0.022	10	0.22
	Optocsatoló	16	6.11	0.07	8	8.96
	DC/DC konverter	1	0.9			0.90
	Rezgőkvarc	1	19.1	0.096	2.1	0.20
	Egyedi ellenállás	20	9.1	0.016	10	3.2
	Ellenállás létrában	16	9.1	0.016	10	2.56
	Választókapcsoló	3	17.1	0.12	1	0.36
	Kondenzátor	6	10.1	0.0064	10	0.38
	LED	1	6.11	0.0012	8	0.01
	Összesen:					17.86
		$17.86*10^{-6}$ 1/óra				
		MTBF=$5.6* 10^{4}$ óra				

Összefoglalás

Jelen példa bemutatásával a megbízhatósági becslés módszerének alkalmazását, az alkalmazás lépéseit kívántuk szemléltetni. A példa jól mutatja azt, hogy a módszer alkalmazása mennyire egyszerű, ugyanakkor a redundanciák figyelembe vétele a módszer alkalmazásánál elsődlegesen csak elhanyagolásokkal (lásd a példában a csatlakozópontok kezelését) lehetséges.

A melléklet irodalomjegyzéke

[4M-1]: Elek L. - Gyenes K. - Pál Gy. - Pesti B. - **Szabó G.**: Korszerű, magas biztonságintegritású ütemadó berendezések a MÁV vonalain. *Vezetékek Világa, Magyar Vasúttechnikai Szemle, 2007/1. szám*, 15-18 oldal, 2007.
[4M-2]: 75 Hz-es biztonsági ütemadó 2. alkalmazás - vonali ütemadó TIMER modul műszaki leírás. Verzióazonosító: E/2006. február 22. PowerQuattro Zrt., 2006.