Dr.sc. Vinko Tomas
Pomorski fakultet u Rijeci
Rijeka, Studentska 2

Pregledni rad
UDK: 629.5.064.5
681.5.09
Primljeno: 05. srpnja 2004.
Prihvaćeno: 12. srpnja 2004.

TEHNIKE TOLERIRANJA KVAROVA U DIJAGNOSTIČKOM SISTEMU

Pri projektiranju dijagnostičkih sistema za brodsko elektroenergetsko postrojenje moramo uskladiti različite ciljeve. Pored ostvarenja predviđenih funkcija, sistem mora imati odgovarajuću pouzdanost, sigurnost, raspoloživost, ekonomičnost, mjerljivost itd. U ovom radu daje se pregled mogućih rješenja kojima se brodski elektroenergetski procesi mogu voditi, uz znatno povoljniju bezotkaznost. Pritom se razmatra postizanje i osiguranje raspoloživosti sistema kao jedne od odlučujućih komponenti ukupne tehničke i korisničke djelotvornosti. Prikazane su tehnike za detektiranje, dijagnozu, ograničavanje, prikrivanje, kompenzaciju i popravak grešaka. Opisani postupci su ilustrirani na primjeru sistema s greškom na aktivnoj jedinici u redundantnom paru. Ove tehnike mogu biti upotrebljavane za projektiranje dijagnostičkog sistema koji tolerira greške. Koje ćemo tehnike koristiti ne ovisi samo o danoj aplikaciji, već i o idejama i filozofiji projektanta.

Ključne riječi: tehnike toleriranja, dijagnostika kvarova, sistemi upravljanja

1. UVOD

Tehnike za izbjegavanje grešaka pomažu spriječavanju otkaza sklopovlja i pogreške u programskoj podršci. One se ostvaruju korištenjem metodologije za projektiranje, kvalitetnom kontrolom, izborom visokokvalitetnih komponenata, periodičnim pregledima projekta, primjenom pravila projektiranja, izradom dokumentacije i njezinim stalnim ažuriranjem. Tehnike koje omogućuju toleriranje kvarova, nose se s otkazima sklopovlja i mogućim pogreškama u softveru. Tehnike koje omogućuju toleranciju kvarova su: uvođenje raznih oblika redundancije, glasanje i tehnike za automatsko obnavljanje ili rekonfiguraciju.

Jedan od nužnih uvjeta za toleriranje kvarova u dijagnostičkom sistemu je

pravovremeno otkrivanje kvara ili greške te, ovisno od željenih osobina sistema, zaustavljanje njegovog rada ili ispravak nastale greške na razne načine. Postoje tri razine na kojima se može primijeniti otpornost na greške, a to su: sklopovska, softverska i razina sistema. Tolerantnost na greške predstavlja načine postizanja pouzdanosti koja je važan zahtjev u mnogim sistemima [3].

Sklopovska otpornost na greške uključuje redundanciju veza prijenosa informacija i izvora energije, repliciranje računalnih modula i drugih resursa. Sama sklopovska otpornost na greške ne garantira visoku raspoloživost sistema. Zato se na drugoj razini mora projektirati dijagnostički softver da se kompenziraju greške, kao što su promjene u programu ili strukturi podataka, radi prijelaznih ili grešaka u projektiranju i razvoju sistema. To se zove softverska otpornost na greške. Takvi mehanizmi su kontrolne točke/restart, memorijski blokovi za obnovu i višestruke verzije programa koji se često koriste na ovoj razini.

Na trećoj razini dijagnostički sistem može imati funkcije koje kompenziraju kvarove u drugim sistemima koji nisu bazirani na računalima, a to se zove sistemska otpornost na greške. Primjerice, dijagnostički softver može detektirati i kompenzirati kvarove u senzorima, izvršnim članovima, trošilima i samim procesnim uređajima u elektroenergetskom postrojenju.

2. UPRAVLJANJE REDUNDANCIJOM

Otpornost na greške se ponekad naziva upravljanje redundancijom [4]. Redundancija predstavlja osiguravanje funkcionalnih mogućnosti koje bi bile nepotrebne u okolišu bez grešaka. Ona je potrebna, ali ne i dovoljna za postizanje otpornosti na greške. Računalni sistem može osigurati takve redundantne funkcije ili izlaze, da je barem jedan rezultat točan u slučaju prisutnosti greške. No, ako korisnik na određeni način ispituje rezultate i izabire one točne, onda on obavlja funkciju otpornosti na greške. Međutim, ako računalni sistem ispravno izabire točan redundantni rezultat umjesto korisnika, onda je on ne samo redundantan već i otporan na greške.

Otpornost na greške uključuje sljedeće postupke: detektiranje, dijagnozu, ograničavanje, prikrivanje, kompenzaciju i popravak grešaka [5].

Detektiranje grešaka je proces utvrđivanja nastanka greške. Tehnike detektiranja grešaka mogu se odvijati on-line i off-line. Kod on-line detekcije sistem je aktivan za vrijeme testiranja, dok kod off-line detekcije on ne može obavljati nikakve funkcije za vrijeme testiranja. Off-line detekcija osigurava integritet prije i vjerojatno za vrijeme rada, ali ne za vrijeme cijelog vremena rada. On-line tehnike moraju garantirati sistemski integritet kroz cijeli proces detekcije. Vrijeme koje protekne dok se detekcija ne obavi zove se prikrivenost greške. Često korišteni mehanizmi detekcije grešaka su:

- Nadgledanje ispravnosti: Jedinica nadgleda ispravnost druge jedinice

tako da očekuje periodične poruke ispravnosti. Jedinica koja se nadgleda mora provjeravati svoju ispravnost i mora slati periodičke informacije o svojoj ispravnosti nadzornoj jedinici. U slučaju gubitka određenog broja uzastopnih poruka o rezultatu ispravnosti, nadzorna jedinica će prijaviti grešku.

- Sat za nadzor: Ovo je hardverski bazirana tehnika nadgledanja koja može detektirati prikrivene neispravnosti u hardverskim ili softverskim modulima. Unutar sistema ugrađeni hardverski sat (programibilni brojač vremena - PBV) u normalnim uvjetima softver periodički restarta. Ako softver uđe u beskonačnu petlju ili se hardver zaglavi, PBV će izazvati hardverski prekid nakon prolaska zadanog vremena. U rutini za obradu ovog prekida pokrenut će se postupci za detekciju i dijagnostiku zbog prevladavanje nastalog stanja.
- Dijagnostika u radu: Često su sklopovski moduli izvedeni tako da dozvoljavaju jednostavne dijagnostičke provjere čak i za vrijeme njihova rada. Te provjere po svojoj prirodi nisu destruktivne, tako da ne utječu na normalan rad modula i sistema.
- Brojač prolaznih grešaka (Transient Leaky Bucket Counter): Kada je sistem u radu, mnoge kratkotrajne prolazne greške može detektirati brojač prolaznih grešaka. Ako brojač detektiranih grešaka prijeđe određenu vrijednost, okidač grešaka aktivirat će odgovarajući dijagnostički program za analizu. Evo nekoliko primjera kratkotrajnih prolaznih grešaka:
    - Sumnjivi prekidi: Nastaju kada se sistemska rutina prekida aktivira, u slučaju kada niti jedan uređaj nije aktivirao taj prekid nego se dogodila prolazna smetnja u sistemu. Hardverska jedinica će aktivirati prekid jedino kada uzastopni lažni prekidi u kratkom vremenskom intervalu, dovedu brojač do neke granične vrijednosti.
    - Sumnjivi okidači grešaka: Kada se okidač greške aktivira, hardverska jedinica je pod sumnjom i nad njom se provodi dijagnostika. Ako dijagnostika zadovolji, jedinica se vraća u sistem a stanje brojača se poveća. Ako se ova procedura učestalo ponavlja, jedinica bi mogla biti u kvaru, ali dijagnostika nije dovoljno iscrpna da bi ustanovila hardversku grešku.
    - Opasna sabirnica: Zbog vanjskih poremećaja, promjene napona napajanja itd., sabirnice u sistemu upravljanja mogu često generirati prekidače grešaka i nakon toga nastaviti pružati uslugu. Ako se to događa prečesto, sabirnica se označava kao opasna i miče se iz sistema. To se čini da bi se izbjeglo preopterećenje sistema, zbog učestale obrade kratkotrajnih grešaka.

Dijagnoza grešaka je proces utvrđivanja uzroka greške podsistema ili komponente s greškom. Dijagnoza grešaka postaje važna kada detekcija grešaka ne može locirati grešku i pružiti ostale informacije o greškama.

Ograničavanje grešaka je proces koji sprječava širenje grešaka s točke nastanka u sistemu na točku gdje može utjecati na uslugu koju dobiva korisnik. Ova se tehnika može upotrijebiti i u hardveru i u softveru. Na primjer, greška se može ograničiti korištenjem krugova detekcije grešaka i promjenama dosljednosti prije izvršenja funkcije ("obostrana sumnja") kao i višestrukim zahtjevima/potvrdama prije izvršenja funkcije.

Ako je jedinica u kvaru, mnogi će se njeni okidači grešaka aktivirati. Glavni zadatak izolacije grešaka je da izvrši korelaciju okidača grešaka te da tako identificira jedinicu s greškom. Ako su okidači grešaka sami po sebi nejasni, procedura izolacije uključuje ispitivanje stanja nekoliko jedinica. Na primjer, ako je protokolarna greška jedina prijavljena greška, sve jedinice na putanji od izvora do odredišta će se ispitati.

Prikrivanje grešaka je proces koji osigurava prolazak točnih vrijednosti do izlaza sistema unatoč neispravnoj komponenti. Prikrivanje grešaka naziva se i statička redundancija koja skriva učinke kvarova tako da redundantna informacija nadjača netočnu informaciju. Primjer prikrivanja grešaka je većinsko glasanje (majority voting). U osnovnom obliku tehnike prikrivanja grešaka ne pružaju njihovu detekciju. Međutim, mnoge se tehnike prikrivanja grešaka mogu proširiti tako da pružaju i on-line detekciju. U suprotnom se tehnike off-line detekcije trebaju koristiti za otkrivanje kvarova.

Kompenzacija grešaka. Ako greška nastane i ako je ograničena na podsistem, potrebno je da sistem kompenzira izlaz podsistema s greškom.

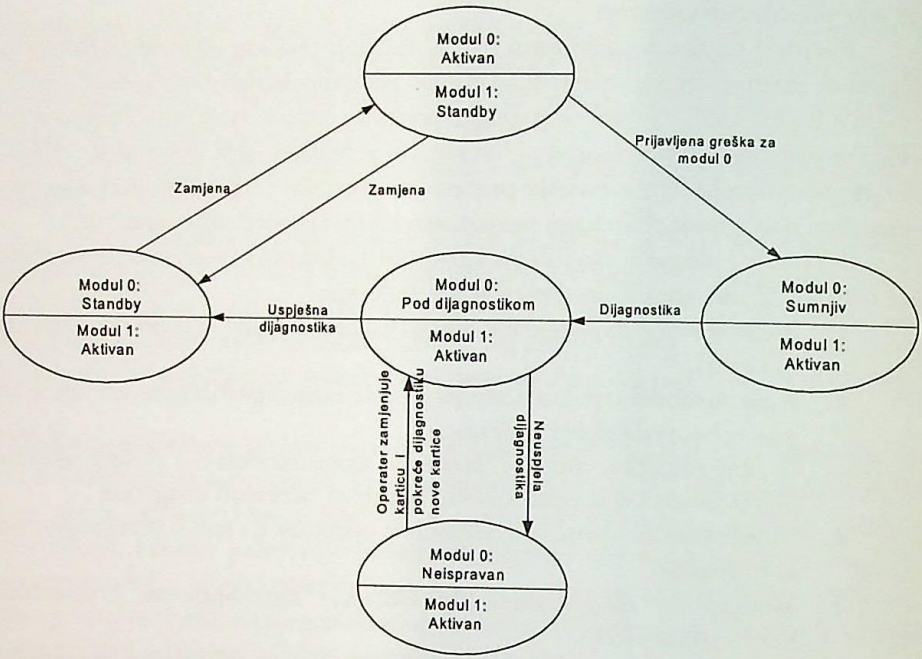
Popravak grešaka je proces s kojim se greške uklanjaju iz sistema. U dobro projektiranim sistemima otpornim na greške, one se ograničavaju prije nego što se raširе do te granice da utječu na uslugu sistema. Dio sistema se ne može koristiti zbog preostalih grešaka. Zbog gubitka dijela resursa sistem se ne bi mogao nositi sa sljedećom greškom, osim ako se ti sklopovi pomoću procesa oporavka ponovno vrate u funkciju, što bi osiguralo nestanak grešaka iz sklopova ili stanja sistema. Popravak sklopa može biti off-line ili on-line, a nakon popravka uređaj ili modul se mora reintegrirati u sistem. Kod on-line popravka reintegracija se mora obaviti bez zastoja u radu sistema. Nakon što se greška detektira i locira, sistem može biti u mogućnosti da izvrši rekonfiguraciju s ciljem zamijene pokvarene komponente ili izoliranja te komponente od ostatka sistema. Komponenta može biti zamijenjena rezervnim dijelom ili se može jednostavno isključiti tako da se performanse sistema degradiraju. To se zove postupna degradacija i predstavlja jednu od tehnika dinamičke redundancije. Nakon detekcije i moguće rekonfiguracije, učinci grešaka se moraju eliminirati. Rad sistema se obnavlja od operacija koje prethode detekciji grešaka. Taj oblik oporavka, koji se često naziva rollback, obično koristi pomoćne potprograme- kontrolne točke i vođenje dnevnika. Kod oporavka, latentnost pogreške postaje važno pitanje zato jer se rollback mora dovoljno vratiti unazad da bi se izbjegli učinci ne detektiranih grešaka koje su se

pojavile prije one detektirane.

Sistem je primoran na ponovni start u slučaju prevelikog oštećenja informacija ili kada se sistem ne može oporaviti od greške. Ponovno startanje može biti "vruće", "toplo" ili "hladno". "Vruće" restartanje, ponovno započinjanje svih operacija od trenutka detekcije greške, moguće je samo ako je nastala šteta obnovljiva. "Topli" restart ponovno pokreće samo neke procese bez gubitaka. "Hladni" restart odgovara cjelovitom ponovnom pokretanju sistema, ako nikakvi procesi nisu preživjeli.

Na slici 1. prikazan je rad sistema s greškom na aktivnoj jedinici u redundantnom paru. Stanja na slici opisujemo sljedećim odrednicama:

1. Pretpostavimo da je modul-0 aktivna jedinica, a modul-1 jedinica u pripremi (standby).
2. Kada se modul-0 pokvari, modul-1 će detektirati grešku pomoću nekog od mehanizama detekcije grešaka.
3. U tom trenutku, modul-1 preuzima ulogu modula-0 i postaje aktivan. Stanje modula-0 se označava kao sumnjivo, očekujući dijagnozu.
4. Sistem aktivira alarm, obavještavajući operatera da radi u neredundantnoj konfiguraciji.
5. Modul-0 se dijagnosticira. To uključuje samodijagnozu i dijagnozu hardverskog sučelja.
6. Ako dijagnostika na modulu-0 prođe, ona će se vratiti u sistem kao standby jedinica. Ako dijagnostika padne, modul-0 se označava kao neispravan i operater se obavještava o neispravnoj kartici.
7. Operater zamjenjuje neispravnu karticu i naređuje sistemu da reintegrira karticu.
8. Sistem provodi dijagnozu na novoj kartici da bi se uvjerio u ispravnost kartice.
9. Nakon što prođe dijagnozu, modul-0 se označava kao standby jedinica.
10. Modul-0 sada počinje nadgledati ispravnost modula-1 koji je trenutačno aktivan.
11. Sistem zaustavlja alarm neredundantne konfiguracije jer je redundancija ponovno uspostavljena.
12. Operater može obnoviti originalnu konfiguraciju, tako da zamijeni stanja modula.



Slika 1. Rukovanje s greškom u redundantnom paru

Mjera uspješnosti otpornosti na greške dijagnostičkog sistema naziva se ***pokrivanje grešaka***, a podrazumijeva vjerojatnost da sistem ne ide u neko od stanja kvara u slučaju nastanka greške. Jednostavne procjene pokrivenosti mjere redundanciju pomoću broja uspješnih redundantnih putanja sistema. Složenije procjene oslanjaju se na činjenicu da svaka greška potencijalno smanjuje otpornost sistema na sljedeće greške. Uobičajeni način procjene je Markovljev model u kojem svaka greška ili popravak prebacuje sistem u nova stanja, od kojih su neka stanja kvara. Implementacija opisanih pravila ovisi o vrsti korištene redundancije.

3. REDUNDANCIJA U DIJAGNOSTIČKOM SISTEMU

Želja da se sistem ostvari s maksimalnom raspoloživosti dovela je do prihvaćanja niza metoda pomoću kojih se postiže cjelovita ili parcijalna strategija proizvodnje otkazno-tolerantnih sistema vođenja. Ugradnjom redundancije (povećana raspoloživost u real-time aplikacijama) neizbježno dolazi do penala koji se odnose na povećanje hardvera, broja ulazno-izlaznih veza, degradaciji odziva kruga sistema i na pitanja vremenske sinkronizacije. Sasvim je razumljivo da se nameće i pitanje cijena višeg stupnja raspoloživosti. No, ako polazimo od kriterija valjanosti broda, odnosno korisničke djelotvornosti, doplata na cijenu osnovne redundantne strukture,

odnosno razlika u cijeni jednostavne sistemske strukture i složene strukture otkazno tolerantnih sistema, ostaje u mjeri koja opravdava cilj. Jednostavne strukture rijetko se primjenjuju u praksi. Kako brodski elektroenergetski sistem ima uočljivu strukturu, pooštreni zahtjevi za ostvarenje kriterija raspoloživosti mogu tretirati samo dio sistema, odnosno onaj podsistem koji je odgovoran za rizično organizirane potprocese ili za one potprocese kod kojih je zahtijevana bezotkaznost rada (FAIL SAFE) visoka.

Pored toga, izbor redundancije ne ovisi samo o sistemu, već i o vrsti očekivanih kvarova. Dodavanje redundancije svim dijelovima neprihvatljivo je i preskupo. Osim toga, niti jedan sistem ne može imati sposobnost toleriranja svih vrsta kvarova [2].

Toleriranje kvarova ostvaruje se oblicima redundancije kao što su: sklopovska, vremenska, redundancija u programskoj podršci i računanju i informacijska redundancija, primjenom kodova sa zalihošću.

Sklopovska redundancija predstavlja postupak za uvođenje ili povećanje redundancije u sistemu, koji se zasniva na dodatnim sklopovima. Umnožavanje sklopova može se primijeniti na svim razinama složenosti sistema od komponenti, funkcijskih modula, podsistema ili cijelog sistema. Udvostručavanje omogućava otkrivanje kvara ili pogreške jednog od modula uspoređivanjem izlaza. Utrostručavanje omogućava otkrivanje neispravnog modula većinskim odlučivanjem, ali i prikrivanje neispravnosti, jer sistem radi ispravno i uz postojanje kvara. Iako je ovaj postupak među najskupljima, on se primjenjuje kod zasnivanja visoko pouzdanih sistema.

Kada se pouzdanost i sigurnost promatraju zajedno može se pokazati da ne mora svako dodavanje modula biti korisno. Jednostavno dodavanje još jednog modula ne mora povećati pouzdanost i sigurnost. Potrebno je dodati najmanje dva modula da bi se istovremeno popravilo jedno i drugo. Ako izlazi individualnih modula ne sadrže neki oblik redundancije. Ako moduli sadrže ugrađenu sposobnost otkrivanja grešaka (samodijagnostiku), dodavanje samo jednog modula može biti dovoljno da se popravi i pouzdanost i sigurnost.

Vremenska redundancija predstavlja oblik redundancije koji se sastoji u ponavljanju nekih operacija prilikom izvođenja programa. Razina primjene može biti od strojnog ciklusa, naredbe, procedure do programa. Uspoređivanjem rezultata nakon ponovljenog izvođenja, može se otkriti kvar ili pogreška. Pri tome se treba imati na umu da, za razliku od ostalih postupaka za povećanje redundancije, ovaj postupak "troši" vrijeme pa se može primijeniti u slučajevima kad ono nije kritično.

Redundancija programske podrške odražava se u postojanju dodatnih dijelova programa ili dodatnom pohranjivanju podataka, a može biti i nekoliko programskih podrški. Ovi postupci postaju sve zanimljiviji zbog sve veće složenosti i veličine programske podrške, pa time i povećane mogućnosti pojavljivanja neispravnosti u njoj. Tehnike mogu biti slične onima za sklopovsku redundanciju, samo se sklopovske cjeline ovdje zamjenjuju programskim. Umnoženi programski moduli mogu se upotrebljavati izvođeni sekvencijalno ili istovremeno uz detekciju neispravnosti nekog od programskih modula pomoću posebnih ispitnih programa i usporedbe rezultata

više modula.

Informacijska redundancija odnosi se na tehnike kodiranja, a primjenjuje se na naredbe, podatke i ostale informacije u sistemu upravljanja. Redundancija se ovdje pojavljuje u obliku dodatnih bitova, koji omogućavaju otkrivanje i/ili ispravljanje pogrešaka. Ovisno o načinu ostvarenja, ovaj oblik redundancije može se promatrati i kao sklopovska redundancija, ako je riješena sklopovljem, ili kao redundancija programske podrške, ako je riješena posebnim programima.

Redundancija u sistemu može biti primijenjena na jedan od načina kao što su [3]: statička, dinamička i hibridna redundancija. Ova je podjela načinjena prema načinu vremenskog djelovanja tehnike za povećanje redundancije. Glavna odlika statičke redundancije je da je sistem s redundantnim dijelovima stalno aktivan, bez promjene građe sistema u slučaju kvara ili pogreške. Primjeri takvih sistema su udvostručeni sistem s usporedbom i utrostručeni sistem s glasanjem. U slučaju kvara na jednom od modula, pogreška će biti otkrivena i po mogućnosti prevladana glasanjem, odnosno prikrivena. Ono što je rečeno za sklopovsku redundancija odnosi se i na statičku. Za razliku od statičke redundancije građa sistema s dinamičkom redundancijom mijenja se nakon otkrivanja kvara ili pogreške. Pri tome se isključuje neispravni dio i uključuje neki novi dio sistema. Budući da za rekonfiguraciju sistema treba neko vrijeme, ovaj oblik redundancije primjenjuje se u sporijim sistemima. Smanjena je osjetljivost na prolazne kvarove i na kvarove sa zajedničkim uzrokom, na koje su sistemi sa statičkom redundancije osjetljivi, zbog istovremenosti rada svih modula. Udruživanjem oba navedena oblika, statičke i dinamičke redundancije, nastaje hibridni oblik kao kompromis povoljnijih svojstava jednog ili drugog oblika redundancije. Svaki od opisanih postupaka ima svoje područje primjene, s određenim prednostima i nedostacima, stoga i ima više zapaženih pojedinačnih rješenja, a uopćavanje je teško izvesti.

4. ZAKLJUČAK

U ovom članku prikazane su tehnike za postizanje tolerancije grešaka. U većini slučajeva kombiniranjem hardverske, informacijske, vremenske i softverske redundancije mogu se postići zahtijevana pouzdanost i tolerancija na pogreške. Jedino mogući ispravan put u ostvarenju procesnih zbivanja jest prethodno sistemsko sagledavanje problema interakcije proces-sistem upravljanja- informativnost - kauzlaritet. Na osnovi takve analize slijedi inženjerski pristup projektiranju procesnog postrojenja i sistema upravljanja. Pri tome se misli da se analizom procesnosti dolazi do spoznaja o upotrebi redundantnih struktura, što zapravo znači da odabrana redundancija na razini nekog procesa ne mora biti identična na razini drugog procesa.

Ovakve strukture karakterizira visoka cijena, a ona je uvjetovana: redundantnim

hardverom; dodatnim komunikacijskim linijama itd.. Primjena ovakvih struktura nalazi opravdanje samo radi upravljanja onim procesima koji zahtijevaju izrazito visok stupanj pouzdanosti, raspoloživosti i sigurnosti. Na brodu uz strukture energetskih procesa, sistem za zaštitu od opasnih stanja mogao bi se svrstati u visoko zahtjevne. Takvih procesa i sistema na ratnom brodu zacijelo ima i među sistemima za vođenje borbe. Upravo identifikacija procesa s obzirom na fizikalnost, namjenu i važnost prethodi strukturiranju sistema upravljanja u pogledu odabira redundantnih struktura. Sam proces ili tehnološko sredstvo ne može se više tretirati samo u radnoj točki ili oko nje. Moć i redundancija omogućavaju da se promjenjivim strukturama upravlja promjenjivom strukturom. Tako će proces biti siguran, a brod kao sistem djelotvoran.

LITERATURA

[1] Tomas, V. Model distribuiranog dijagnostičkog sistema brodskih elektroenergetskih postrojenja: doktorska disertacija, Rijeka, Pomorski fakultet, 2003.

[2] Grimmelius, H.T. The use of first principle modelling for faulty behaviour prediction in the design stage, INEC 2002: Glasgow, The marine engineer in the electronic age, 2002.

[3] Proctor, F.M. Open architecture controllers, Presentation to the panel on manufacturing Process Control, National Research Council, Washington, D.C., May, 1997.

[4] Ying, S. Fault tolerance computing-draft, Carnegie Mellon University, 1999., www.ece.cmu.edu/~koopman/des_s99/fault_tolerant/index.html

[5] Hwee Tou Ng, University of Texas at Austin; Model-based, multiple-fault diagnosis of dinamic, continuous physical devices, IEEE Expert, december 1991.

[6] Vlahinić, I.; Tomas, V. Pristup osnivanju sistema vođenja električne centrale sa integriranim dizel generatorima, Zbornik radova Pomorskog fakulteta u Rijeci, 9(1995), str.87-95.

Summary

TECHNICAL TOLERANCE OF FAULTS IN THE SYSTEM OF DIAGNOSIS

During the designing of diagnostic systems for the ship's electric power propulsion the coordination of different objectives is necessary. In addition to the realization of the anticipated functions, the system must have a corresponding reliability, security, availability, economy, quantification, etc. This paper presents a survey of possible solutions by which the ship's electric power processes can be managed with a considerably more satisfactory failure-free operation. In this connection consideration is given to achieving and securing the availability of the system as one of the crucial components of the overall technical and user efficiencies. The techniques for detection, diagnosis, limitation, hiding, compensation and repair of faults are presented. The described procedures are illustrated on the example of a system with a fault in an active unit in a redundant pair. These techniques can be used for designing a diagnostic system that tolerates faults. As regards which techniques should be used, does not depend only on the given application, but likewise on the ideas and philosophy of the designer.

Key words: technical tolerance, failure diagnoses, control system

Faculty of Maritime Studies Rijeka
Studentska 2, 51000 Rijeka
Croatia