FONDUL SOCIAL EUROPEAN
Investeşte în oameni!
Programul Operaţional Sectorial pentru Dezvoltarea Resurselor Umane 2007 – 2013
Proiect POSDRU/6/1.5/S/19 – *Pregătirea competitivă a doctoranzilor în domenii prioritare ale societăţii bazate pe cunoaştere*

UNIVERSITATEA **POLITEHNICA** DIN BUCUREŞTI

Facultatea de Automatică şi Calculatoare

Catedra de Calculatoare

Nr. Decizie Senat 211 din 15.09.2011

TEZĂ DE DOCTORAT

- rezumat -

Soluţii de asigurare a calităţii sistemelor software de monitorizare şi control a staţiilor electrice

Quality assurance solutions for electrical substations monitoring and control software systems

Autor: Ing. Victor Ursianu

COMISIA DE DOCTORAT

Preşedinte	Prof. Dr. Ing. Dumitru Popescu	de la	Facultatea de Automatică şi Calculatoare, Universitatea POLITEHNICA din Bucureşti
Conducător de doctorat	Prof. Dr. Ing. Florica Moldoveanu	de la	Facultatea de Automatică şi Calculatoare, Universitatea POLITEHNICA din Bucureşti
Referent	Prof. Dr. Ing. Sergiu Stelian Iliescu	de la	Facultatea de Automatică şi Calculatoare, Universitatea POLITEHNICA din Bucureşti
Referent	Prof. Dr. Ion Smeureanu	de la	Facultatea de Cibernetică, Statistică şi Informatică Economică, Academia de Studii Economice din Bucureşti
Referent	Prof. Dr. Ing. Mat. Dumitru Dan Burdescu	de la	Facultatea de Automatică, Calculatoare şi Electronică, Universitatea din Craiova

Bucureşti, 2011

Cuprins

1. Introducere

1.1. Problema calităţii software

Problema calităţii sistemelor software de monitorizare şi control din domeniul energetic este de mare complexitate.

Calitatea sistemului depinde de calitatea tuturor componentelor sale, hardware şi software. Corectitudinea funcţionării unui sistem de timp real depinde atât de rezultatele calculelor cât şi de momentul în care sunt ele disponibile. În cazul sistemelor embedded, calitatea depinde de întregul ansamblu calculator-echipament.

Aplicaţiile în timp real din prima generaţie rulau pe sisteme monoprocesor, problemele ce le rezolvau fiind relativ simple şi nepresupunând algoritmi sofisticaţi sau prelucrări foarte complexe. În domeniul energetic, aplicaţiile constau din programe privind optimizarea centralelor electrice astfel încât să se reducă preţul de cost al energiei.

În ceea ce priveşte calitatea software pentru aceste sisteme, vom folosi următorii termeni:

- **Defect (Defect):** o problemă software legată de comportarea sa externă sau caracteristicile sale interne, o anomalie în produsul software (“bug”).
- **Cădere (Failure**): incapacitatea unui sistem sau componentă de a-şi realiza funcțiile cerute conform specificațiilor de performanţă (IEEE 610.12, 1990).
- **Greșeală** (**Fault**): un pas, proces incorect, o definiție de date incorectă într-un program (IEEE 610.12).
- **Eroare** (**Error**): o acțiune umană care produce un rezultat incorect (IEEE 610.12).

Astfel, căderea poate fi interpretată ca o abatere comportamentală de la cerințele utilizatorului sau de la specificația produsului în timpul operării, greșeala ca fiind condiție existentă în software care cauzează o cădere iar eroarea reprezintă o acțiune umană absentă sau incorectă care are ca efect injectarea unor greșeli în produsul software.

1.2. Importanţa asigurării calităţii software

Datorită experienţei dobândite în domeniul calităţii software şi în ultimii ani în domeniul monitorizării echipamentelor electrice primare şi apoi al staţiilor electrice în totalitate, din punct de vedere al aparatajului primar, consider că aceste sisteme de monitorizare şi control pot fi îmbunătăţite. Există mai multe aspecte ce pot fi studiate, inclusiv pe partea de proiectare, dezvoltare şi mentenanţă a acestor sisteme.

Mai întâi de toate, trebuie conştientizată importanţa domeniului energetic. Acesta este un domeniu fundamental al dezvoltării economice naţionale şi internaţionale. Fără energie,

nu se poate dezvolta nimic. Pentru dezvoltarea conceptului de Smart Grid (Reţea Inteligentă) sunt puse la dispoziţie fonduri financiare uriaşe la nivelul Uniunii Europene şi nu numai.

Există o părere unanimă a specialiştilor în acest domeniu că domeniul energetic şi viitorul Smart Grid nu poate funcţiona fără monitorizarea parametrilor echipamentelor electrice din cadrul staţiilor electrice. Staţiile electrice reprezintă zona intermediară cea mai importantă între producătorul de energie şi utilizatorul final. În funcţie de tipul lor constructiv, staţiile electrice se împart în mai multe categorii. În cadrul tezei de doctorat, m-am referit la staţiile electrice de transformare aparţinând reţelei electrice naţionale de transport a energiei electrice. Soluţiile de asigurare a calităţii software prezentate, împreună cu modelele matematice propuse pot fi folosite şi la alte tipuri de staţii electrice.

Calitatea unui produs software este dată de "capacitatea sa de a putea fi utilizat eficient, efectiv şi confortabil, de către un set de utilizatori, pentru un set de scopuri, în condiţii specificate".

Asigurarea calităţii software a sistemului este foarte importantă din necesitatea de a avea un produs fiabil, cu cât mai puţine defecte şi căderi. În cazul cel mai rău, situaţia poate genera un dezechilibru energetic în zona respectivă şi pagube materiale importante pentru consumatori.

1.3. Structura şi obiectivele tezei

Următoarele capitole ale tezei sunt:

- Capitolul 2, intitulat "Asigurarea calităţii software" descrie pe scurt metricile software: metrici de dimensiune, complexitate, calitate a produsului şi respectiv metrici interne ale procesului. Sunt prezentate soluţii de asigurare a calităţii software, care presupun utilizarea de tehnici şi metode de prevenire a injectării defectelor precum şi tehnici de eliminare a defectelor şi izolare a acestora.
- Capitolul 3, intitulat "Analiza aspectelor de calitate ale unui sistem software de monitorizare şi control a unei staţii electrice" prezintă arhitectura generală a unui astfel de sistem. Este prezentat sistemul de monitorizare şi control EMCSIT care a fost utilizat ca studiu de caz. Sunt propuse tehnici de asigurare a calităţii software.
- Capitolul 4, intitulat "Studiu de caz: asigurarea calităţii software pentru sistemul de monitorizare şi control EMCSIT" prezintă arhitectura şi funcţionalitatea sistemului EMCSIT de monitorizare şi control a unei staţii electrice din România. Sunt descrise soluţiile şi exemple de aplicare a acestor soluţii pentru asigurarea şi îmbunătăţirea calităţii software a acestui sistem. Pentru îmbunătăţirea procesului de testare a IED-

urilor (Intelligent Electronic Device) incluse în acest sistem este propusă dezvoltarea unui simulator software ce generează pachete de date şi le trimite pe interfaţa serială a calculatorului server.

- Capitolul 5, intitulat ”Standardul IEC61850” descrie limbajul SCL de configurare a unei staţii electrice şi aspecte generale ale standardului IEC61850.
- Capitolul 6. intitulat “ Utilizarea unor modele matematice pentru estimarea fiabilității sistemelor de monitorizare și control a stațiilor electrice” prezintă modelul de distribuţie Rayleigh şi rezultatele aplicării acestui model asupra ratelor de defectare obţinute pentru sistemul EMCSIT folosit ca studiu de caz. Este prezentat modelul matematic al lanţurilor Markov şi o propunere de implementare a acestui model prin care se poate prezice momentul când acest sistem va cădea. Se prezintă modelarea unei reţele Bayesiene având drept noduri tip părinte, IED-urile ce monitorizează echipamentele electrice. Cu ajutorul informaţiilor privind rata de defectare/cădere a acestora, se propune estimarea fiabilităţii sistemului.
- Capitolul 7, intitulat “ Aplicaţie software pentru estimarea fiabilității unui sistem de monitorizare şi control a unei staţii electrice” descrie componentele incluse în aplicația dezvoltată pentru estimarea fiabilităţii sistemului: *Parser SCL* şi *Calcul BN*. Utilizând informaţii privind rata de defectare/cădere a IED-urilor, se calculează probabilitatea de defectare a întregului sistem de monitorizare şi control.
- Capitolul 8, intitulat “Concluzii, contribuţii proprii şi planuri de cercetare pentru viitor” prezintă concluziile cercetării şi contribuţiile proprii originale dezvoltate şi aplicate în scopul asigurării calităţii sistemelor software de monitorizare şi control a staţiilor electrice. Sunt propuse direcţii de cercetare pentru viitor.

Obiectivele tezei sunt următoarele:

- Identificarea modalităţilor de îmbunătăţire a procesului de dezvoltare software pentru asigurarea unui nivel ridicat de fiabilitate şi a unei mentenabilităţi crescute pentru un sistem de monitorizare şi control a unei staţii electrice.
- Crearea si optimizarea unor instrumente software de testare a modulelor sistemului.
- Alegerea unor modele matematice utile pentru estimarea fiabilităţii la nivel de componentă precum şi la nivelul întregului sistem.
- Dezvoltarea unor module software pentru estimarea fiabilităţii sistemului.

2. Asigurarea calităţii software

2.1. Calităţile generale ale unui produs software

O metrică software este o măsură a unei proprietăţi pentru un artefact software. O metrică software trebuie să fie cuantificabilă şi să poată fi aplicată unei caracteristici a produsului software.

O aplicaţie software care prezintă calitate internă ridicată este uşor de modificat, uşor de extins cu noi facilităţi şi uşor de testat. Software-ul cu o calitate internă scăzută este greu de înţeles, greu de schimbat şi dificil de extins.

Calitatea software externă este o măsură a modului în care sistemul în ansamblul său îndeplineşte cerinţele beneficiarului.

Încercările de standardizare a terminologiei referitoare la calitatea produselor software au condus la standardul ISO 9126.

2.2. Metrici pentru evaluarea calităţii software

Sunt următoarele categorii de metrici:

a. Metrici de dimensiune (size metrics)

Metricile de dimensiune sunt folosite pentru aprecierea aspectelor legate de calitate, productivitate şi de estimare a costurilor.

- **Linii de cod (LOC, Lines Of Code)**
- **Puncte funcţiune (FP - function points)**

b. Metrici de complexitate

Sunt definite: complexitatea codului (complexitatea ciclomatică), complexitatea proiectării, complexitatea integrării unui ansamblu de module.

c. Metrici de calitate a produsului

Calitatea produsului reprezintă "totalitatea caracteristicilor ce îi conferă acestuia aptitudinea de a satisface nevoile utilizatorului" [Swq9-6].

Calitatea din punctul de vedere al producătorului reprezintă conformitatea cu cerinţele. Din punctul de vedere al utilizatorului, calitatea reprezintă conformitatea cu aşteptările utilizatorului. Sunt două nivele ale metricilor de calitate a produsului software :

- Metrici de calitate intrinsecă;
- Metrici orientate către client;

d. Metrici interne ale procesului (in-process metrics)

Metricile interne ale procesului au rolul de a exprima starea software-ului din punct de vedere practic şi a susţine eliminarea defectelor înainte de livrarea aplicaţiei software către beneficiar.

2.3. Metode şi tehnici de asigurare a calităţii sistemelor software

2.3.1. Prevenirea injectării defectelor în software

Prevenirea injectării defectelor prin blocarea sau eliminarea surselor de eroare are la bază următoarele:

- Eliminarea anumitor surse de eroare, cum ar fi: comunicarea ambiguă, neînţelegerea cerinţelor, etc.
- Prevenirea sau blocarea greşelilor prin corectarea sau blocarea directă a erorilor umane.

2.3.2. Tehnici de eliminare a defectelor

Principalele tehnici de eliminare a defectelor, înainte ca produsul software să fie folosit de către utilizatorul final sunt:

- Activităţile de inspecţie: se detectează şi se elimină greşelile din codul sursă, documentele de proiectare şi specificare;
- Testarea: se elimină greşelile pe baza căderilor constatate în timpul execuţiilor programului;

2.3.3. Tehnici de toleranță la defecte

Datorită complexităţii şi dimensiunii multor sisteme software actuale, metodele de prevenire şi reducere a defectelor nu pot elimina toate defectele: numărul de teste necesare ar putea fi prea mare. Tehnicile de toleranţă la defecte pornesc de la premiza existenţei defectelor în diferite componente ale unui sistem, scopul lor fiind de a menţine sistemul în funcţiune în cazul apariţiei unui defect.

Principalele tehnici de toleranţă la defecte sunt:

2.3.3.1. Utilizarea blocurilor de recuperare (recovery blocks)

Utilizând procesoare din ce în ce mai puternice şi mai rapide, putem repeta anumite sarcini de calcul într-un termen stabilit, fără a afecta grav performanţa sistemului. În acest caz, putem folosi blocuri de recuperare în mod repetat pentru a stabili puncte de control (checkpoints) şi a repeta paşii de calcul atunci când pot apărea sau sunt observate probleme în timpul operării.

Conform [Swq9-9], un bloc de recuperare este executat prin efectuarea fiecărei variante pe rând, începând cu varianta principală, până când pentru o anumită variantă, testul

de acceptanţă este satisfăcut. Executarea fără erori a unei variante este urmată de evaluarea făcută prin testul de acceptanţă. În cazul în care testul nu a fost trecut, va fi semnalată o stare de eroare, la care sistemul răspunde prin restaurarea programului în starea de dinaintea intrării în varianta principală. Execuţia continuă apoi cu următoarea variantă, dacă aceasta există. Dacă, totuşi toate variantele au fost încercate şi niciuna nu a trecut testul de acceptanţă, va fi semnalată o stare de eroare către blocul de recuperare.

2.3.3.2. Folosirea NVP (N-version programming)

Domeniul ingineriei software include metode prin care se poate cuantifica şi îmbunătăţi calitatea programelor. Una dintre soluţiile studiate este „programarea cu N versiuni" (NVP) [Swq9-7].

Bug-urile software sunt persistente: aflat în aceleaşi condiţii, programul se va comporta în acelaşi fel. Tehnicile de votare sunt neputincioase dacă toate componentele returnează aceeaşi eroare în acelaşi timp. Programarea cu N versiuni se realizează prin executarea în paralel a N programe diferite, scrise de echipe diferite de programatori, dacă e posibil, folosind instrumente şi tehnologii diferite. Toate cele N programe rezolvă aceeaşi problemă, dar în moduri diferite. Folosind o astfel de strategie, tehnica votării poate funcţiona în cazul programelor.

2.3.3.3. Self-checking

Metoda self-checking (software cu auto-verificare) nu este o metodă riguros descrisă în literatura de specialitate [Swq9-8], ci mai degrabă o metodă ad-hoc folosită în unele sisteme importante.

Software-ul cu auto-verificare include controale suplimentare, inclusiv mai multe puncte de verificare (checkpoint) şi metode de recuperare (rollback) introduse în sisteme tolerante la defecte sau sisteme critice. Alte metode includ taskuri separate, care acţionează în stivă (heap), găsind şi corectând defecte de date. În timp ce auto-verificarea nu poate fi o metodologie riguroasă, aceasta s-a dovedit a fi surprinzător de eficientă.

2.3.3.4. Reconfigurare şi reîntinerire

Reconfigurarea şi reîntinerirea sunt variante complementare pentru software-ul tolerant la defecte. Reconfigurarea este reactivă în timp ce reîntinerirea este proactivă.

Reconfigurarea software-ului poate utiliza resurse redundante pentru recuperarea în timp real, în timp ce consideră, în mod dinamic, influenţa mai multor factori (serviciile sistemului de operare, încărcarea procesorului, memoria, etc.).

Reîntinerirea este o abordare nouă pentru remedierea erorilor software datorate vechimii software-ului. Aceasta poate fi văzută ca o soluţie preventivă şi proactivă ce este utilă pentru împiedica fenomenul de îmbătrânire a software-ului.

Tehnica implică oprirea rulării software-ului la anumite momente de timp, „curăţarea" proceselor interne şi repornirea lui. Curăţarea proceselor interne ale unui software presupune refacerea spaţiului disponibil (garbage collection), curăţarea tabelelor kernel-ului sistemului de operare, reiniţializarea structurilor de date interne, etc.

2.3.3.5. BASE

BASE (Byzantine Abstract Specification Encapsulation) este o tehnică de toleranţă la defecte, bazată pe BFT (Byzantine Fault Tolerance). BFT presupune dezvoltarea unui serviciu care să tolereze un comportament arbitrar faţă de replicarea defectelor, ca de exemplu comportamentul cauzat de un bug software sau un atac informatic.

Spre exemplu, dacă în cadrul unei aplicaţii software, o funcţie depinde de rezultatele alteia şi acea funcţie furnizează rezultatele cu o mică abatere (eroare) atunci ea se va propaga prin intermediul celei de-a doua funcţii şi se va ajunge în final la un rezultat total eronat.

2.3.4. Securitatea şi integritatea transmisiei datelor

Deoarece sistemele de monitorizare şi control a staţiilor electrice, utilizează o arhitectură client-server, un alt aspect important de asigurare a calităţii este securitatea transmisiei datelor între server şi clienţi.

Pentru o creştere a securităţii transmisiei datelor se poate folosi protocolul IPv6. Datorită dimensiunii mari a spaţiului unei adrese IPv6, scanarea aleatoare după sisteme ce sunt vulnerabile este complet inutilă deoarece durează foarte mult numai scanarea spaţiului adreselor IPv6 alocate furnizorilor de servicii Internet. Scanările targeted, deşi nu sunt uşoare, sunt încă posibile astfel că măsuri de securitate precum cele folosite pentru IPv4 sunt încă necesare [Net9-2].

Este absolut necesară asigurarea integrităţii datelor transmise între echipamentele hardware de monitorizare şi aplicaţiile software ce achiziţionează informaţii de la acestea (aplicaţiile tip server). Una dintre metodele propuse este utilizarea CRC - Cyclic Redundancy Check (Control Redundant Ciclic).

3. Analiza aspectelor de calitate pentru un sistem de monitorizare şi control a unei staţii electrice

3.1. Arhitectura generală pentru un sistem de monitorizare şi control a unei staţii electrice

Un sistem de monitorizare şi control a unei staţii electrice presupune existenţa unor echipamente electronice care monitorizează parametrii echipamentelor (denumite IED: Intelligent Electronic Device), unul sau mai multe calculatoare (servere) care centralizează informaţiile monitorizate şi le salvează într-o baza de date şi unul sau mai multe calculatoare pe care se executa o aplicaţie client cu rol de prezentare a parametrilor monitorizaţi.

Un astfel de sistem are, în general, o arhitectură client-server (fig. 3.1.1.). Aplicaţiile server achiziţionează de la IED-uri, în timp real valorile parametrilor monitorizaţi ai echipamentelor electrice, le prelucrează şi apoi le salvează într-o bază de date locală sau centrală. Aplicaţia client achiziţionează informaţii din baza de date şi le prezintă utilizatorului (personalul din staţia electrică) printr-o interfaţă grafică prietenoasă (GUI – Graphical User Interface).

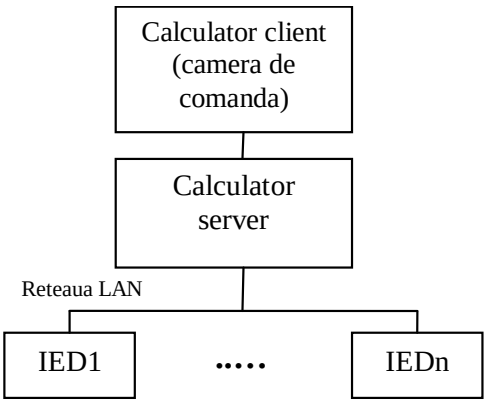

Fig. 3.1.1. Arhitectura generală a unui sistem de monitorizare şi control a unei staţii electrice

3.2. Sisteme existente de monitorizare şi control a staţiilor electrice

În prezent, în domeniul monitorizării şi controlului staţiilor electrice, există câteva mari companii internaţionale de renume ce se ocupă de dezvoltarea hardware şi software a unor astfel de sisteme.

Sistemele de monitorizare pe care le-am considerat ca fiind reprezentative sunt cele dezvoltate de:

- General Electric: sistemul iSM&D (Integrated Substation Monitoring and Diagnostic);
- ABB: sistemul SMS510 (Substation Monitoring System);
- Siemens: sistemul iSCM (Integrated Substation Condition Monitoring);
- AREVA: sistemul PACiS (Protection, Automation&Control Integrated Solution);

Precizez că niciuna dintre aceste companii nu a implementat încă un sistem de monitorizare şi control complet a unei staţii electrice din România. Un astfel de sistem necesită investiţii financiare foarte mari atât din partea beneficiarului cât şi din partea producătorului.

Producătorii de echipamente electrice sunt avantajaţi în cazul în care dezvoltă un astfel de sistem deoarece pot integra şi controla mai uşor propriile echipamente.

Am avut ocazia să lucrez la un proiect pentru un astfel de sistem de monitorizare şi control complex, într-o staţie electrica aparţinând reţelei de transport a energiei electrice din România, dezvoltat de compania Nova Industrial.

Nova Industrial este o companie românească ce a dezvoltat sistemul EMCSIT (Echipament pentru Monitorizarea Complexă a Staţiilor de Înalta Tensiune) pentru monitorizarea şi controlul staţiilor de înalta tensiune [Smg9-1].

EMCSIT este un sistem complex de monitorizare on-line a unei staţii electrice. Din punct de vedere hardware, sistemul este alcătuit din mai multe IED-uri (Intelligent Electronic Device) poziţionate în fiecare cabină de relee din cadrul staţiei, care sunt conectate la senzori şi traductori montaţi pe echipamentele electrice din staţie şi transmit informaţiile achiziţionate de la acestea către serverele locale.

Sistemul de monitorizare include aplicaţii de tip server ce se conectează la echipamentele de monitorizare (IED-uri); ele achiziţionează valorile parametrilor monitorizaţi şi le salvează într-o bază de date centrală, instalată pe un calculator server central. Acesta este accesat de aplicaţiile client EMCSIT din reţeaua locală.

Sistemul de monitorizare şi control EMCSIT include următoarele tipuri de aplicaţii: EMCSIT Server, EMCSIT Client şi EMCSIT Staţie:

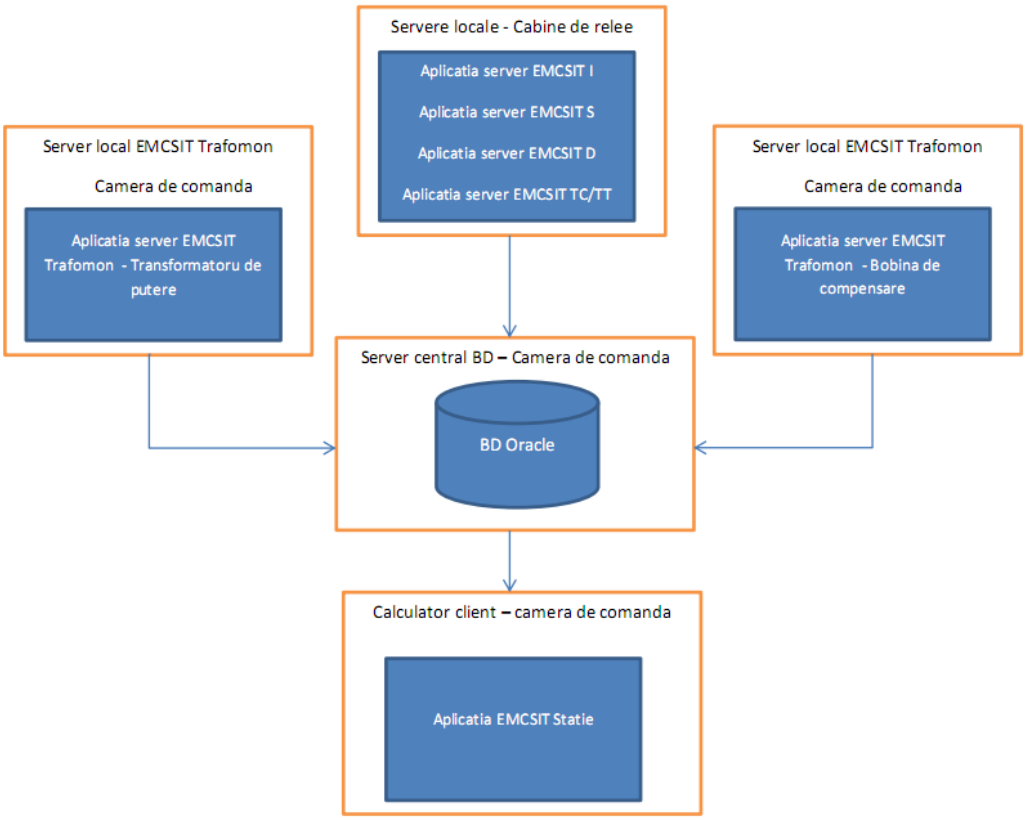

Fig.3.2.1. Arhitectura sistemului EMCSIT

Aplicaţiile "EMCSIT Server" care sunt instalate pe serverele locale:

- Sunt specifice fiecărui tip de echipament electric primar dintr-o staţie electrica (transformator de putere, întreruptor, separator, descărcător, transformator de măsură de curent sau tensiune);
- Asigură achiziţia datelor de la echipamentul de monitorizare EMCSIT (IED), prelucrarea acestor date conform cu specificaţiile specialiştilor tehnologi din domeniu şi salvarea acestor date în baza de date locală;

Aplicaţiile "EMCSIT Client" care sunt instalate pe calculatorul client din camera de comandă (sau pe orice calculator client) sunt dezvoltate pentru fiecare tip de echipament electric, şi au următoarele componente:

- Componenta de vizualizare a ultimelor date achiziţionate şi înregistrate de server: EMCSIT - Client.
- Componenta de vizualizare şi analiză a evenimentelor înregistrate de echipamentul de monitorizare IED (pentru întreruptoare, separatoare şi transformatoare de măsură), EMCSIT - Grafice Evenimente. Un eveniment pentru un echipament electric reprezintă o variaţie bruscă a parametrilor monitorizaţi.

- Componenta de vizualizare a istoricului datelor măsurate şi înregistrate de către serverele locale: EMCSIT Istoric.

Aplicaţia "EMCSIT Staţie" prezintă schema monofilară completă pentru staţia electrică, afişând simboluri animate pentru echipamentele electrice primare monitorizate şi afişează valorile parametrilor monitorizaţi privind fiecare astfel de echipament.

3.3. Smart Grid

Tendinţa internaţională, şi recent naţională, este de a crea reţele inteligente (Smart Grids) ce includ sisteme de monitorizare şi control în domeniul energetic.

Majoritatea lumii ştiinţifice mondiale acceptă că "Reţelele/Reţeaua Inteligentă" vine de la termenul "Smart Grids/Grid". Indiferent de definiţia utilizată, o Reţea Inteligentă include un sistem automat de monitorizare şi control în timp real a lanţului producţie - consumator final de energie.

Analizând modelul american şi european privind realizarea Reţelelor Inteligente, se constată că una dintre verigile importante privind implementarea Smart Grid-urilor, ca parte componentă a infrastructurii instalaţiilor electrice, este staţia electrică. Pentru a contribui la realizarea acestui concept în România, a fost realizat proiectul EMCSIT.

3.4. Probleme care pot cauza căderi ale sistemului

Sistemul de monitorizare şi control poate cădea atunci când una dintre componentele sale, hardware sau software cade sau se defectează.

Aplicaţiile software pot avea următoarele cauze ale căderilor:

- Neprotejarea secţiunilor din codul sursă la generarea de excepţii;
- Conectarea eronată sau neconectarea la IED-uri;
- Apariţia unor bucle infinite în secvenţele de achiziţionare de informaţii sau evenimente;
- Netratarea tuturor condiţiilor care pot să apară în procesul de funcţionare a echipamentelor electrice, etc.

În concluzie, problemele care pot cauza proasta funcţionare a unui sistem de monitorizare şi control a unei staţii electrice pot fi de mai multe feluri, de la întârzierea transmisiei datelor, afişarea eronată a datelor, până la nefuncţionarea unei componente software sau chiar a întregului sistem.

Informaţiile obţinute cu ajutorul sistemului de monitorizare şi control sunt utile la dispeceratul energetic teritorial (DET) respectiv naţional (DEN) iar o întârziere de raportare a unui defect al echipamentului electric monitorizat poate afecta echilibrul energetic pe o anumita zonă.

4. Studiu de caz: soluţii pentru asigurarea calităţii sistemului de monitorizare şi control EMCSIT

4.1. Particularităţile sistemului

EMCSIT este un sistem complex de monitorizare on-line a unei staţii electrice [Urs8-3]. Arhitectura sistemului EMCSIT este de tip client-server.

Din punct de vedere hardware, sistemul de monitorizare este alcătuit din mai multe IED-uri (Intelligent Electronic Device) poziţionate în fiecare cabină de relee din cadrul staţiei electrice, care sunt conectate la senzori şi traductori montaţi pe echipamentele electrice din staţie şi transmit informaţiile achiziţionate de la acestea către serverele locale.

Sistemul de monitorizare, din punct de vedere informatic, include calculatoare (denumite în continuare servere) care sunt conectate la echipamentele de monitorizare (IED-uri) şi achiziţionează valorile parametrilor monitorizaţi, pe care le stochează într-o bază de date locală Oracle. Pe lângă aceste servere, există un server central care efectuează sincronizarea cu celelalte servere şi preia informaţiile pentru a fi accesate de aplicaţiile client EMCSIT din reţeaua locală.

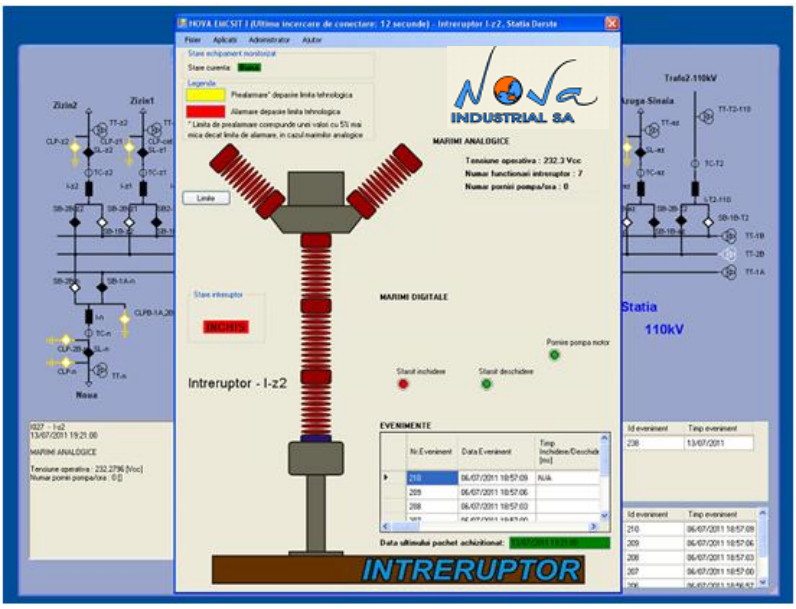

Fig. 4.1.1. Aplicaţia EMCSIT Staţie ce prezintă schema monofilară a staţiei şi integrarea aplicaţiei EMCSIT client pentru monitorizarea unui întreruptor

Deoarece un server local se conectează la mai multe IED-uri, va trebui să ruleze câte o instanţă de aplicaţie server pentru fiecare IED conectat. Pentru a preveni rularea a mai

multor ferestre de aplicaţii server, fiecare conectându-se la câte un IED, a fost dezvoltată o aplicaţie denumită „EMCSIT SuperServer”. Aceasta rulează câte o instanţă de aplicaţie server aferentă fiecărui IED, transparent pentru utilizator, utilizând mai multe thread-uri (fire de execuţie). În cazul în care există erori pentru achiziţia informaţiilor de la un IED, thread-ul respectiv va fi abandonat, urmând ca operaţiunea să se reia la următorul interval de achiziţie. În cadrul sistemului folosit ca studiu de caz, intervalul de achiziţie este de 1 minut. Astfel, la fiecare minut, aplicaţia EMCSIT SuperServer interoghează toate IED-urile care sunt conectate la serverul local şi achiziţionează informaţii privind parametrii monitorizaţi ai echipamentelor electrice.

După achiziţionarea pachetului de date de la IED, informaţiile sunt prelucrate şi salvate în baza de date, instalată pe serverul central aferent câte unei substaţii din componenţa staţiei electrice. În cazul în care se produce un eveniment (închidere/deschidere întreruptor sau separator, supracurenţi, supratensiuni, descărcare, etc.) pentru un echipament electric monitorizat, se va deschide un alt thread pentru achiziţionarea, prelucrarea şi salvarea lui în baza de date, utilizând un modul software dedicat, EMCSIT Evenimente.

În perioada de dezvoltare a sistemului, au fost colectate informaţii privind numărul de defecte descoperite la nivelul fiecărei aplicaţii din componenţa sistemului. Pe baza acestor informaţii, a fost calculată rata de defectare a fiecărei componente software a sistemului.

Printre erorile software observate, menţionez:

- netratarea tuturor posibilităţilor ce pot apărea în exploatare, în calcule, ce pot duce la rezultate eronate;
- netratarea împărţirilor la 0;
- autoscalarea temporizată greşit a mărimilor ce duce la intrarea în bucle infinite;
- proiectarea şi implementarea software greşită, astfel încât la anumite erori să se depăşească memoria alocată sau stiva şi să se întrerupă funcţionarea normală, etc.

Este propusă pentru viitor, ca soluţie pentru remedierea acestor tipuri de erori/defecte, implementarea unui watchdog hardware (mecanism de reiniţializare a IED-ului în cazul blocării), tratarea cât mai multor excepţii posibile în software, validarea internă a rezultatelor înainte de a trimite comenzi către echipamentul electric monitorizat.

4.2. Dezvoltarea unui simulator de IED-uri pentru testarea sistemului

Dezvoltarea unui sistem de monitorizare şi control a unei staţii electrice include atât componente software cât şi hardware şi necesită un efort financiar şi uman uriaş.

Este necesar ca un astfel de sistem să fie cât mai fiabil. Pentru aceasta se efectuează teste independente pentru componentele software şi apoi teste de acceptanţă în staţie, având toate subsistemele conectate şi funcţionale. Am considerat necesară dezvoltarea unui simulator software pentru IED-uri, datorită numeroaselor avantaje: testare mult mai completă, obținerea unui produs mult mai fiabil, scurtarea perioadei de teste de acceptanţă în staţia electrică, etc.

Acest simulator de IED-uri transmite pe interfaţa serială pachete de date ce pot fi preluate de către aplicaţiile tip server EMCSIT.

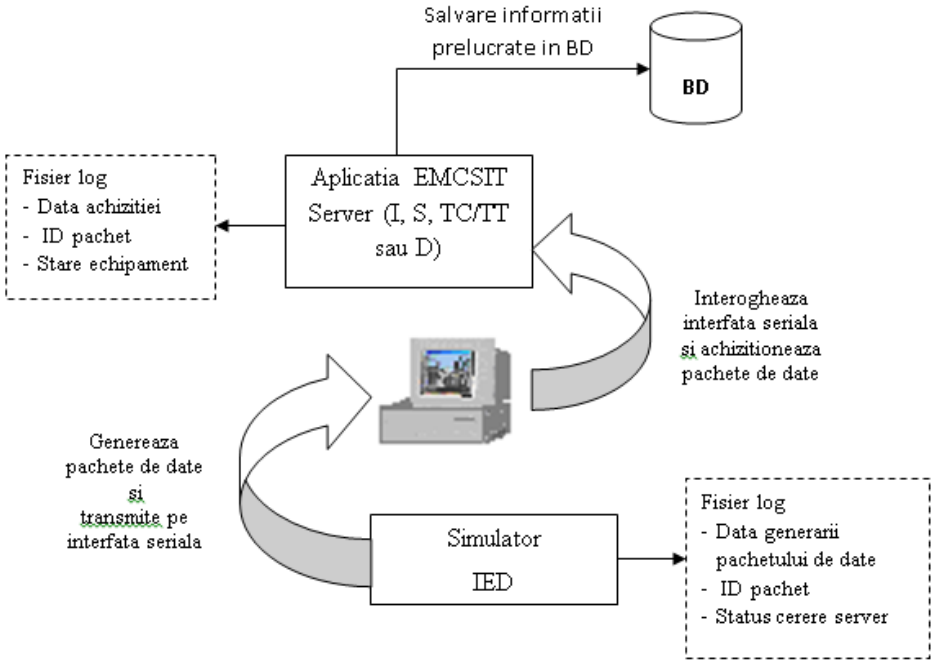

Fig. 4.2.1. Ansamblul simulator IED – aplicaţie server EMCSIT

Pachetele de date transmise de către simulator sunt generate pe baza unor mărimi de intrare specifice echipamentelor electrice conectate la IED. Mărimile de intrare diferă de la un tip de IED la altul. Totodată, protocolul de comunicaţie între IED şi server nu este acelaşi pentru toate tipurile de IED. Este necesară o etapă de configurare în care tester-ul alege tipul de IED ce va fi simulat, precizează numărul mărimilor monitorizate de IED și specifică domeniul de valori al fiecărei mărimi monitorizate şi limitele de alarmare.

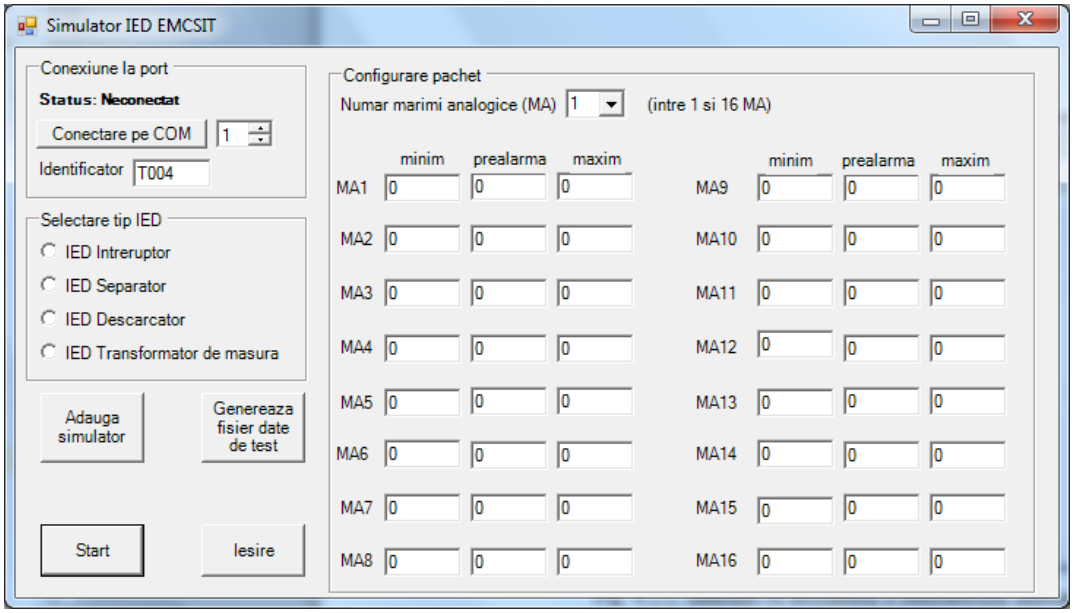

Fig. 4.2.2. Fereastra de configurare a simulatorului de IED-uri

Simularea poate fi pornită pentru IED-ul care a fost configurat sau se poate adăuga alt IED, datele pentru IED-ul curent fiind salvate într-un fişier text de configurare.

Simulatorul are posibilitatea adăugării mai multor IED-uri virtuale datorită implementării protocolului de comunicare între IED-uri – Daisy Chain. IED-urile sunt conectate la un calculator server într-o configuraţie Daisy-Chain prin conectarea fiecărui IED la alt IED, nu prin conectarea individuală, a fiecărui IED direct la server. Numai ultimul IED din această înlănţuire se conectează direct la server. Fiecare IED are un identificator unic, comunicarea făcându-se pe aceeaşi magistrală în funcţie de valoarea acestuia.

În momentul în care se porneşte simulatorul, la intervale predefinite de timp, vor fi generate pachete de date şi trimise pe interfaţa serială pentru achiziţia de către server.

Simulatorul generează fişiere log ce conţin informaţii privind data generării pachetului de date, id-ul acestuia şi dacă a fost achiziţionat cu succes sau nu de către aplicaţia server. În acest moment, simulatorul are implementat un protocol de comunicaţie proprietar.

4.3. Generarea cazurilor de test– Metoda Pairwise testing

Generarea datelor de test prin analiza domeniilor variabilelor de intrare conduce la foarte multe cazuri de test: vectorii de test reprezintă combinaţii ale valorilor selectate pentru variabilele de intrare.

Pentru testarea aplicaţiilor de tip server, simulatorul generează date de test pornind de la specificaţiile mărimilor monitorizate: valoarea minimă, valoarea maximă şi valoarea de prealarmare. Utilizarea *Pairwise testing* [Swq9-5] asigură faptul că fiecare combinaţie

posibilă dintre valorile selectate pentru fiecare pereche de variabile de intrare este acoperită prin cel puţin un test. Studii empirice au arătat că metoda poate conduce la detectarea a aproximativ 70% din defectele existente în software. De exemplu pentru 6 mărimi analogice, fiecare cu 6 valori de test posibile, utilizând metoda Pairwise testing se obţin 540 de cazuri de test, în loc de 6^6 (= 46656) cazuri, rezultate din considerarea tuturor combinațiilor posibile.

Simulatorul pe care l-am dezvoltat generează cazuri de test prin metoda Pairwise testing, pe care le transmite aplicaţiei server. Pot fi modificate atât intervalul de timp la care sunt transmise pachetele de date test cât şi numărul de mărimi analogice, simulatorul putând genera pachete de date cu maxim 16 mărimi analogice.

Utilizând fişierele log generate atât de simulator cât şi de aplicaţia server, se pot face corelări pentru a observa dacă au fost achiziţionate toate datele de test şi starea echipamentului pentru acele valori, astfel putându-se verifica uşor corectitudinea calculelor efectuate în cadrul aplicaţiei server.

4.4. Asigurarea securităţii sistemului EMCSIT

Actualele tehnici de securitate, implementate deja în cadrul sistemului de monitorizare EMCSIT, includ:

- autentificare prin utilizator/parolă pentru accesul la aplicaţia EMCSIT Staţie;
- asignarea unică de adrese IP pentru calculatoarele care fac parte din sistem;
- utilizarea de chei hardware HASP pentru asigurarea securităţii accesului la aplicaţia EMCSIT Staţie;

Protocolul de comunicaţie utilizat între serverele locale şi serverul central este TCP/IP iar securitatea transmisiei datelor este cea implementată şi inclusă în acest protocol. Se intenţionează ca pe viitor să se configureze reţeaua pentru a fi utilizat protocolul IPv6.

4.5. Implementarea unor tehnici de toleranţă la defecte

4.5.1. Folosirea de Blocuri de Recuperare

Tehnica folosirii blocurilor de recuperare este utilizată în cadrul proiectului EMCSIT la achiziţia de evenimente (acţionări ale diverselor echipamente electrice - de ex. deschidere sau închidere întreruptor). Aplicaţia server interoghează fiecare IED conectat pentru a achiziţiona lista de evenimente generate în cadrul staţiei şi înregistrate de IED-uri în memoria internă. Aceasta listă este comparată cu informaţiile existente deja în baza de date pentru a identifica noile evenimente. Atunci când s-a identificat un eveniment nou este transmis id-ul (identificatorul) acestuia împreună cu id-ul IED-ului corespunzător către modulul de achiziţie evenimente. În cazul în care acest modul cade din diverse motive

(eşuare achiziţie pachete de date pentru eveniment de la IED, eşuare salvare în baza de date, etc.), sunt prevăzuţi paşi anteriori unde aplicaţia EMCSIT Server se poate întoarce şi reîncepe procesul de achiziţie eveniment.

4.5.2. Utilizarea tehnicilor de duplicare

Tehnicile de duplicare sunt utilizate în următoarele situaţii:

1) În cazul în care unul dintre echipamente nu mai funcţionează sau comunicaţia cu el eşuează, aplicaţia care comunică cu echipamentul trebuie să atenţioneze utilizatorul şi să încerce să obţină date echivalente de la alte echipamente.

2) În cazul în care legătura între serverele locale ce achiziţionează şi înregistrează datele de la echipamente de monitorizare nu este posibilă, în momentul reluării acesteia, aplicaţiile EMCSIT Server trebuie să achiziţioneze şi să afişeze informaţiile înregistrate în memoria internă a IED-urilor, privind parametrii monitorizaţi.

4.5.3. Utilizarea tehnicilor de reconfigurare şi reîntinerire (reconfiguration and rejuvenation)

Sistemul EMCSIT poate fi astfel configurat pentru ca fiecare server local sau serverul central să aibă posibilitatea la un moment dat, în funcţie de mai mulţi factori, să se reiniţializeze. Reconfigurarea ţine seama de încărcarea procesorului, memoria internă liberă insuficientă, porturile de comunicaţie blocate, probleme de interfaţă de reţea, etc.

Soluţia o reprezintă alocarea unor resurse suplimentare pentru funcţionarea în continuare a sistemului. În cazul sistemului EMCSIT ce rulează pe o platforma Microsoft Windows, o soluţie extremă este programarea la un moment dat a unei reporniri pentru serverul respectiv pe care rulează aplicaţia.

Reîntinerirea reprezintă o tehnică ce ia în considerare faptul că aplicaţia ”EMCSIT SuperServer” sau ”EMCSIT Staţie” poate intra într-o buclă infinită sau să atingă un maxim al resurselor utilizate. În cazul unei reporniri accidentale sau voite, aplicaţia respectivă va rula automat în momentul încărcării sistemului de operare pe server sau calculatorul client.

5. Standardul IEC61850

5.1. Introducere

Staţiile de transformare reprezintă componente cheie ale reţelei electrice, facilitând transportul şi distribuţia energiei electrice. Au un rol vital în monitorizarea şi controlul fluxului de energie electrică şi asigură interconectarea între companiile de producere electricitate, reţelele de transport şi distribuţie şi consumatorii finali.

Sistemele de automatizare ale staţiilor electrice fac posibilă monitorizarea şi controlul în timp real şi ajută la maximizarea eficienței, siguranţei şi a integrităţii datelor.

Ultimii ani au adus dezvoltări semnificative în ceea ce privește standardele ce definesc comunicaţia la nivel de staţii electrice – probabil cel mai important pas în acest sens a fost publicarea standardului IEC 61850. Tot mai mulţi producători şi-au adaptat deja produsele pentru a fi conforme cu noul standard sau chiar au dezvoltat produse noi pornind de la modelul de date pe care-l definește standardul. Noul standard este tot mai mult impus şi de către diverşi beneficiari. Viitorul în comunicaţii la nivel de staţii electrice este deja prefigurat de IEC 61850.

Standardul IEC61850 conţine un set de documente care se axează pe următoarele aspecte majore: un model funcţional al domeniului de aplicare pentru SA (Substation Automation - automatizările din staţia electrică) – partea a 5-a, un model de date pentru SAS (Substation Automation System - sistemul de automatizări din staţie), protocoalele de comunicaţii şi serviciile aferente – partea a 7-a şi părţile 8 şi 9, un limbaj de configurare a staţiei bazat pe XML (SCL – Substation Configuration Language) – partea a 6-a.

IED-ul reprezintă, conform IEC61850 orice echipament ce include unul sau mai multe procesoare (microcontrollere) cu posibilitatea de a primi sau trimite date/control de la sau către o sursă externă.

Acest standard de comunicaţii promite să revoluţioneze automatizările din staţiile electrice cu mesaje peer-to-peer foarte rapide, date structurate şi orientate-obiect.

Este destinat să asigure un singur protocol pentru o întreagă staţie, să implementeze un format obişnuit pentru descrierea staţiei, să faciliteze modelarea datelor necesare staţiei, să definească serviciile de bază necesare pentru a transfera datele prin diferite protocoale de comunicaţie deschise şi să permită interoperabilitatea între echipamente ale diverşilor producători.

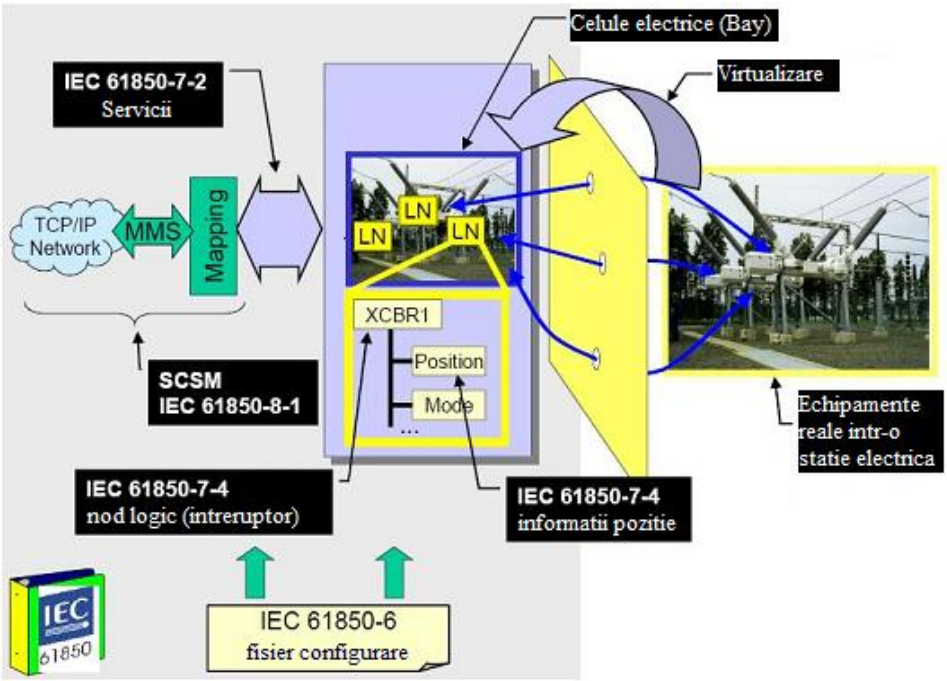

Fig. 5.1.1. Modelarea conceptuală conform standardului IEC61850 [Iec9-10]

5.2. Limbajul SCL

Substation Configuration Language (SCL – limbajul de configurare a staţiei) este limbajul şi formatul de reprezentare specificat de IEC61850 pentru descrierea configuraţiei echipamentelor dintr-o staţie electrică. Acesta include reprezentarea datelor modelate şi a serviciilor de comunicaţie specificate de documentele standardului IEC61850-7-X. Descrierea completă a SCL şi detaliile sunt specificate în documentul standard IEC 61850-6 [Iec9-3]. Include reprezentarea datelor pentru echipamentele din staţie, funcţiile asociate fiecărui echipament - reprezentate ca noduri logice, sisteme şi capabilităţi de comunicaţie. Reprezentarea completă a datelor în limbajul SCL oferă posibilitatea interoperabilităţii echipamentelor dintr-o staţie electrică prin schimb de fişiere.

Marele avantaj al utilizării IEC61850 este interoperabilitatea între IED-uri aparţinând diverşilor producători din domeniu. Limbajul are o convenţie de denumire standard a datelor, echipamentele sunt descrise automat prin intermediul limbajului SCL, permite modelarea virtuală a echipamentelor logice şi oferă un limbaj comun de configurare a echipamentelor.

6. Utilizarea unor modele matematice pentru estimarea fiabilităţii sistemelor de monitorizare şi control a staţiilor electrice

6.1. Modelul de distribuţie Rayleigh

Modelul Rayleigh este un model parametric, care se bazează pe o distribuţie statistică specifică. Atunci când parametrii distribuţiei statistice sunt estimaţi pe baza datelor dintr-un proiect software, pot fi făcute previziuni referitoare la rata de defectare a produsului software.

Modelul Rayleigh este unul dintre cele mai folosite modele matematice pentru modelarea şi predicţia ratelor de defectare a produselor software în timp ([Urs8-1], [Urs8-2]).

Am aplicat modelul Rayleigh pe datele înregistrate în timpul dezvoltării aplicaţiilor server pentru cele 5 tipuri de IED-uri prezentate în studiul de caz EMCSIT (cap. 4).

Datele primare reprezintă numărul de defecte înregistrate în fiecare lună pentru aplicaţiile tip server destinate monitorizării întreruptoarelor (IED1), separatoarelor (IED2), descărcătoarelor (IED3), transformatoarelor de măsură de curent/tensiune (IED4) respectiv transformatorului de putere și bobinei de compensare (IED5).

Pe abscisă este precizată luna în care au fost colectate informaţiile privind numărul de defecte iar pe ordonată este reprezentată valoarea funcţiei de densitate a probabilității. S-a obţinut următorul rezultat pentru aplicaţia tip server IED1:

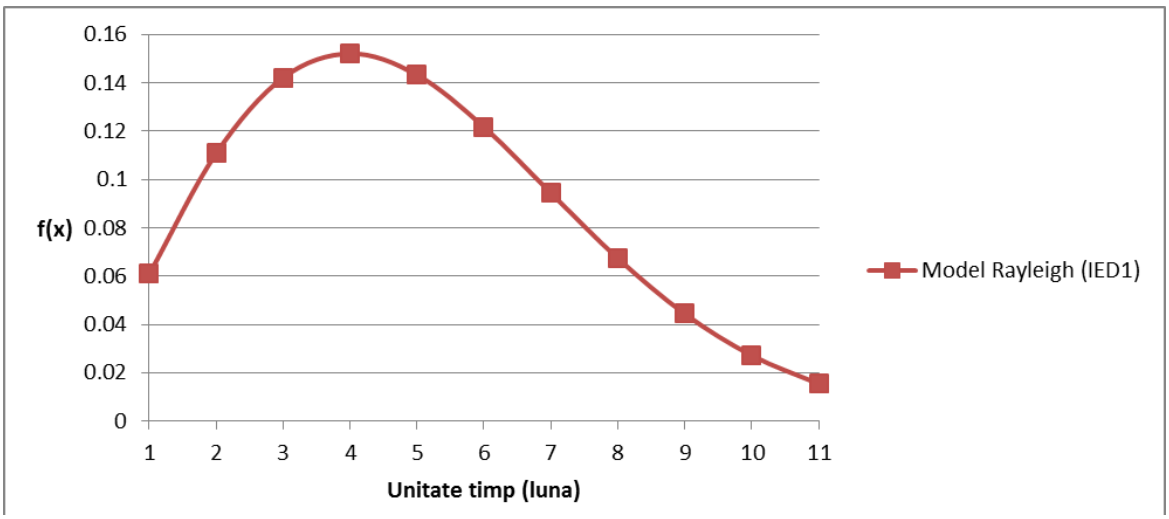

Fig. 6.1.1. Aplicarea modelului de distribuţie Rayleigh pentru aplicaţia tip server IED1

În urma aplicării modelului Rayleigh pentru aplicaţiile server asociate celor 5 tipuri de IED-uri, s-a constatat că pentru IED3, IED4 şi IED5 rata descoperirii defectelor pe

parcursul dezvoltării a fost bună, astfel încât la sfârşitul perioadei de testare rata s-a stabilizat şi a avut o valoare mică. Modelarea cu ajutorul curbei Rayleigh a ratei defectelor pentru serverul IED1 indică faptul că la sfârşitul perioadei de testare rata a scăzut dar nu s-a stabilizat încă, fiind posibilă descoperirea unor noi defecte în perioada imediat următoare. Pentru serverul IED2, la sfârşitul perioadei de testare rata defectelor a fost mare, numărul de defecte rămase în software a fost de asemenea mare, indicând faptul că testele efectuate au fost insuficiente.

Concluzia generală în urma aplicării modelului Rayleigh este că în situaţia în care s-ar fi utilizat simulatorul software de la începutul proiectului, se puteau descoperi mai multe defecte înainte de instalarea în staţie.

6.2. Modelul matematic al lanţului Markov

În teoria probabilităţilor, un model Markov este un proces stochastic ce posedă proprietatea Markov. Această proprietate se referă la faptul că stările viitoare ale unui sistem modelat printr-un lanţ Markov depind de cele prezente şi sunt independente de cele trecute.

Pe baza experienţei personale de dezvoltare şi utilizare a unor sisteme software de monitorizare şi control, propun ca un astfel de sistem să fie asociat la un moment dat cu una dintre următoarele stări:

1. Stare bună (S1): sistemul în ansamblu funcţionează conform specificaţiilor;

2. Stare acceptabilă (S2):

- există probleme la achiziţionarea pachetelor de date. Sunt pachete ratate (neachiziţionate de către aplicaţiile server) la anumite intervale de timp;
- rarele erori nu presupun încă intervenţia personalului din staţie;

3. Stare proastă (S3):

- sistem indisponibil majoritatea timpului;
- există pachete de date ratate foarte des;
- întârzieri în actualizarea datelor afişate;
- este necesară intervenţia utilizatorului pentru repornirea uneia sau mai multor aplicaţii din sistem;
- poate fi necesară chiar repornirea calculatorului client, serverului sau IED-ului/IED-urilor;

4. Stare inacceptabilă (S4):

- majoritatea sau chiar toate componentele sistemului nu mai funcţionează sau nu funcţionează conform specificaţiilor;

o sistemul nu mai este disponibil;

Pentru estimarea stării curente a sistemului se pot folosi informaţii din fişierele log, informaţii din bazele de date ale sistemului, informaţii privind resursele de sistem utilizate de către servere, etc. Metoda propusă de mine pentru estimarea stării curente şi pentru calculul timpilor de tranziţie între stări se bazează pe analiza fişierelor log.

În cazul sistemului de monitorizare şi control EMCSIT folosit ca studiu de caz şi prezentat în capitolul 4, fişierele log create de aplicaţiile server conţin informaţii privind data achiziţiei fiecărui pachet de date, id-ul pachetului, dacă acesta a fost valid şi starea echipamentului electric monitorizat. Analizând fişierul log, se poate observa dacă au lipsit pachete de date din cele care trebuiau să fie achiziţionate la intervale de timp prestabilite (de ex. la fiecare minut).

Observând comportamentul sistemului pentru o perioadă suficientă de timp, am constatat că trecerea dintr-o stare în alta este strâns corelată cu numărul de pachete de date pierdute (neachiziţionate).

Astfel, propun următoarea clasificare a stărilor sistemului în cele 4 stări discrete:

- În cazul în care există mai puţin de 1 pachet de date ratat (neachiziţionat de către aplicaţia server) pe zi atunci sistemul se află în starea S1.
- Dacă există cel mult 1 pachet de date ratat la fiecare 6 ore sau în medie cel mult 4 pachete ratate pe zi atunci sistemul se află în starea S2. În aceeaşi stare sistemul se încadrează şi dacă are maximum două pachete de date ratate consecutiv.
- Dacă sunt cel mult 24 pachete ratate pe zi adică în medie 1 pachet de date la fiecare oră sau mai mult de 3 pachete de date ratate consecutiv atunci sistemul se află în starea S3.
- Pentru situaţiile mai grave decât cele descrise anterior, sistemul se află în starea S4.

Se poate utiliza modelul Markov pentru a stabili timpii de tranziție între stările sistemului de monitorizare și control și respectiv când va ajunge sistemul în starea inacceptabilă [Urs8-20]. Prin introducerea în modelul Markov al sistemului de monitorizare și control, a două stări intermediare între cea bună și cea inacceptabilă, este posibilă intervenția promptă pentru remedierea defectelor înainte ca sistemul să ajungă în starea de funcționare inacceptabilă.

6.3. Reţele Bayesiene

6.3.1. Modelul matematic

O reţea Bayesiană este un graf orientat fără cicluri ce reprezintă relaţiile probabilistice între variabilele unei mulţimi. Variabilele mulţimii sunt figurate drept noduri, notate cu $\{X_1,X_2,...X_n\}$ iar relaţiile între ele sunt reprezentate prin arce:

• nodul părinte reprezintă variabila care determină schimbarea variabilei fiu;

• nod fiu este variabila care suferă influenţa variabilei părinte;

Intensitatea şi modul în care se manifestă influenţele între variabile sunt redate prin tabelul de probabilităţi condiţionate al fiecărui nod în parte. Acesta ne permite să calculăm valorile probabilităţilor pentru variabila aleatoare asociată nodului, în funcţie de valorile părinţilor.

Fiecare nod are un tabel de probabilităţi condiţionate asociat lui. Probabilităţile condiţionate sunt bazate pe informaţiile din trecut. O probabilitate condiţionată este scrisă matematic ca P(x|p1,p2,...pn) şi reprezintă probabilitatea ca variabila X să fie în starea x dacă părintele P1 se află în starea p1, P2 se află în starea p2,..., respectiv Pn se află în starea pn.

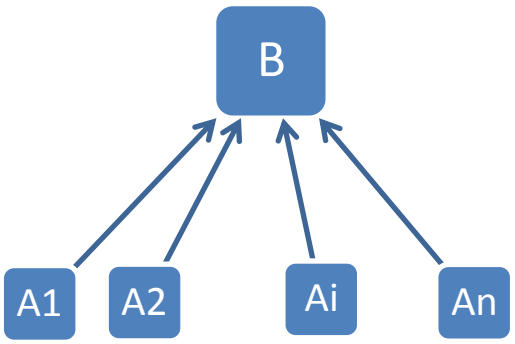

Fig. 6.3.1.1. Exemplu de reţea Bayesiană

În cazul unei reţele Bayesiene, cum ar fi cea din fig. 6.3.1.1., pentru a calcula probabilitatea de defectare a nodurilor părinte notate cu Ai, se aplică formula lui Bayes pentru calculul probabilităţii condiţionate:

$$P(A_i|E = defect) = \frac{P(A_i) \times P(E = defect|A_i)}{\sum_{j=1}^{n} P(A_j) \times P(E = defect|A_j)}$$

unde n reprezintă nodurile ce influenţează direct nodul i, i=1,...,n.

Probabilitatea ca nodul fiu, B, să se defecteze este următoarea:

$$P(B) = \sum_{i=1}^{n} P(A_i) \times P(E = defect|A_i)$$

6.3.2. Modelarea unui sistem de monitorizare şi control a unei staţii electrice conform standardului IEC61850 printr-o reţea Bayesiană

Reţelele Bayesiene pot estima probabilitatea de defectare a unui sistem software pe baza istoricului ratelor de defectare ale modulelor constituente. Ne propunem să modelăm printr-o reţea Bayesiană, un sistem de monitorizare şi control a unei staţii electrice [Urs8-6].

Aceasta ne va permite să calculăm probabilitatea de defectare a fiecărui nod din sistem precum şi a întregului sistem. Pentru calculele referitoare la probabilităţile de defectare a nodurilor (IED-urilor) cât şi a sistemului de monitorizare şi control trebuie achiziţionate informaţii privind numărul de defecte descoperite într-o anumită perioadă de timp, pentru fiecare IED. Fiecare nod fiu este caracterizat de o anumită rată de defectare (număr de defecte într-o perioadă de timp). Reţeaua Bayesiană, fiind un graf orientat, suportă orice relaţie între noduri. În funcţie de relaţiile/influenţa între noduri, se calculează probabilitatea de defectare a acelui nod.

6.3.3. Studiu de caz: modelarea sistemului EMCSIT printr-o reţea Bayesiană

Sistemul EMCSIT poate fi modelat prin mai multe reţele Bayesiene. Astfel, se poate construi câte o reţea Bayesiană pentru fiecare server local, ce poate fi descris ca un subsistem de monitorizare şi control; vor fi 5 reţele Bayesiene cu ajutorul cărora se va putea calcula probabilitatea de defectare a fiecărui subsistem de monitorizare şi control.

Se poate construi o reţea Bayesiană pentru serverul central, ce va avea drept noduri părinte serverele locale din cabinele de relee, notate cu Server local CR1 (B1), Server local CR2 (B2), Server local CR3 (B3), Server local CR4 (B4) respectiv Server local CR5 (B5) adică subsistemele de monitorizare şi control. Vor fi utilizate informaţiile privind probabilităţile de defectare ale serverelor locale, calculate anterior.

Toate cele 5 noduri reprezentând serverele locale sunt conectate în reţea la un server central de baze de date denumit în reţeaua Bayesiană ca fiind nodul SS1. Funcţionarea acestui nod va fi influenţată de nodurile B care sunt independente între ele.

În final, reţeaua Bayesiană prin care se va putea calcula probabilitatea de defectare a întregului sistem de monitorizare şi control a staţiei electrice va avea drept noduri părinte: serverul central, IED-urile ce monitorizează transformatorul de putere respectiv bobina de compensare (de tipul IED5) și calculatorul ce rulează aplicaţia client din camera de comandă.

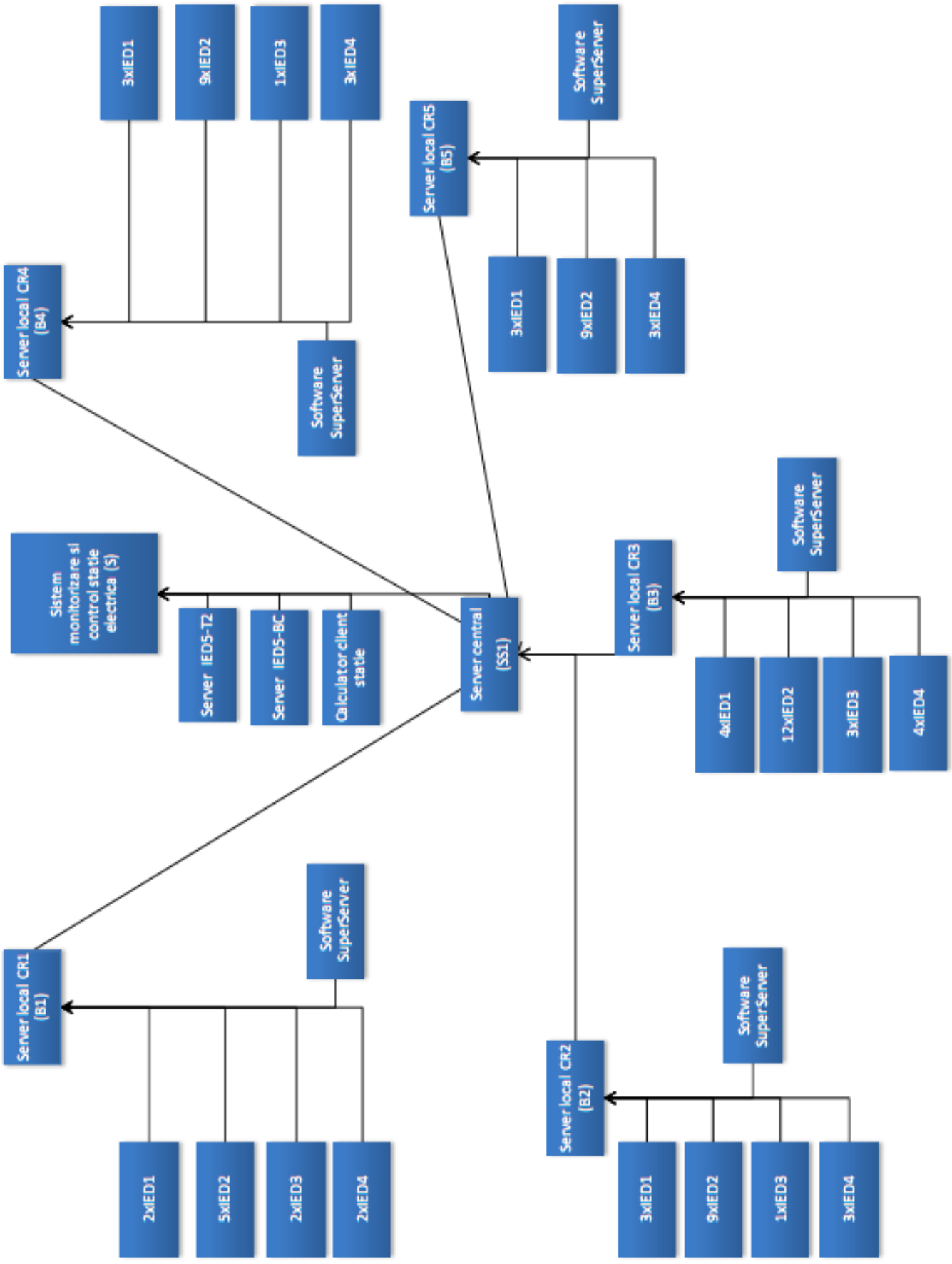

Fig. 6.3.3.1. Reprezentarea sistemului EMCSIT printr-o reţea Bayesiană

7. Aplicaţia software pentru estimarea fiabilităţii unui sistem de monitorizare şi control a unei staţii electrice

Aplicaţia software permite modelarea unui sistem de monitorizare şi control a unei staţii electrice printr-o reţea Bayesiană şi evaluarea acesteia în scopul estimării probabilităţii de defectare a fiecărui nod al reţelei precum şi a întregului sistem. Configuraţia staţiei trebuie să fie descrisă într-un fişier în format SCL, conform standardului IEC61850. Fişierul poate fi creat cu un editor special (comercial) sau manual.

Aplicaţia este compusă din modulele *Parser SCL* şi *Calcul BN*.

7.1. Modulul Parser SCL

Pentru modelarea unei staţii electrice conform standardului IEC61850 s-a folosit limbajul SCL (Substation Configuration Language) şi s-a utilizat aplicaţia Visual SCL [Iec9-11]. Această aplicaţie permite utilizatorului să salveze schemele de staţie în formatul standard tip XML specific IEC61850 sub forma unor fişiere cu extensia .SCD.

Pentru utilizarea informaţiilor prezente în fişierele .SCD am dezvoltat un modul software, denumit *Parser SCL.* Acesta citeşte informaţii din fişierele .SCD scrise în limbajul de configurare a unei staţii electrice – SCL (IEC61850) şi furnizează ca date de ieşire, structuri de date ce pot fi apoi definite ca noduri ale unei reţele Bayesiene. Datele de ieşire sunt folosite ulterior de aplicaţia *Calcul BN*, pentru estimarea fiabilităţii sistemului.

Nodurile şi informaţiile asociate lor, reprezentând datele de defectare ale IED-urilor, vor folosi la calculele privind probabilitatea de defectare a sistemului de monitorizare şi control.

7.2. Modulul Calcul BN

Informaţiile obţinute, respectiv echipamentele electrice şi IED-urile asociate se folosesc în modelarea matematică a reţelei Bayesiene ca noduri ale sistemului de monitorizare şi control a staţiei electrice. După ce au fost făcute conexiunile între IED-uri şi echipamentele electrice, aplicaţia de calcul a reţelei Bayesiene preia lista de IED-uri şi oferă utilizatorului posibilitatea completării ratelor de defectare pentru acestea.

Fluxul completării informaţiilor în cadrul aplicaţiei software de modelare a reţelei Bayesiene este prezentat în paşii următori:

1. Se rulează modulul software *Parser SCL* ce va furniza o parte din datele de intrare ale aplicaţiei. Acestea sunt nodurile reţelei Bayesiene bazată pe schema staţiei electrice conform IEC61850:

- Noduri reprezentând IED-urile;
- Noduri de tip PC, definite de utilizator, ce reprezintă serverele locale din cabinele de relee şi respectiv serverul central de baze de date;
- Nodul de tip HMI (Human Machine Interface) ce reprezintă calculatorul client din camera de comandă ce rulează aplicaţia software tip client;

2. Există posibilitatea adăugării de noi noduri pentru reţeaua Bayesiană, pe lângă cele obţinute din fişierul de configurare a staţiei electrice.
3. Sunt definite nodurile de tip părinte (IED-urile) şi nodul fiu (subsistemul de monitorizare şi control a staţiei electrice - serverul local):

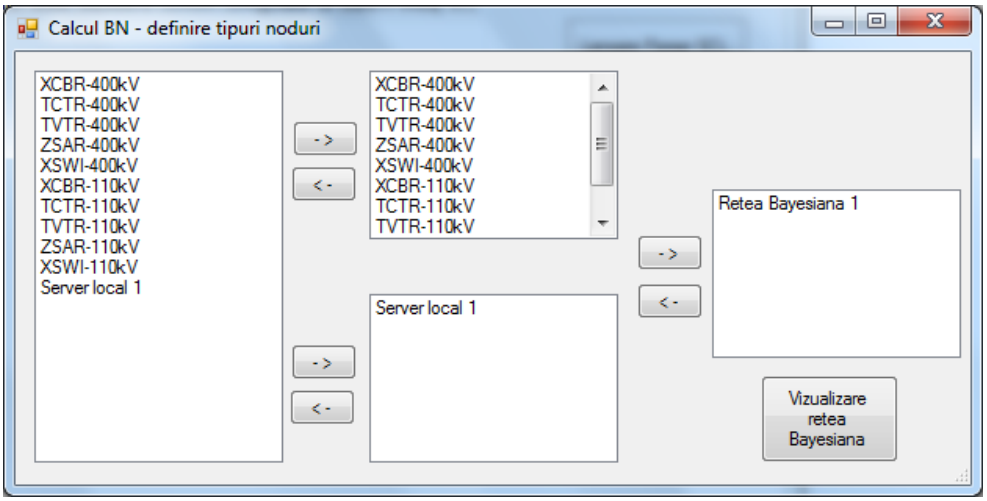

Fig. 7.2.1. Definirea tipurilor de noduri pentru reţeaua Bayesiană.

4. Pentru calculul fiabilităţii fiecărui IED, se încarcă informaţii privind numărul de defecte constatate de-a lungul perioadei de testare (rata de defectare), pentru fiecare nod al reţelei Bayesiene – IED_1,..., IED_n. În situaţia în care nu sunt disponibile date privind defectele înregistrate pentru un anumit IED, se poate seta manual probabilitatea a priori de defectare.
5. Aplicaţia calculează probabilitatea de defectare a sistemului de monitorizare şi control precum şi probabilitatea de defectare a posteriori a fiecărui IED dar şi a subsistemelor (în cazul în care au fost definite).

   În studiul de caz, aceste subsisteme sunt reprezentate de serverele locale din cabinele de relee.

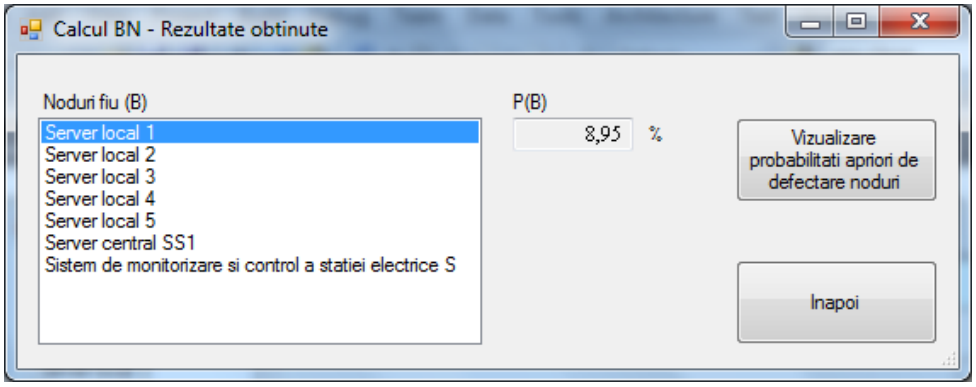

Fig. 7.2.2. Fereastra de afişare a rezultatelor finale

7.3. Rezultate obţinute

Pentru estimarea fiabilităţii unui sistem de monitorizare şi control utilizând metoda reţelelor Bayesiene, a fost folosită o configuraţie de staţie electrică de transformare aparţinând reţelei de transport a energiei electrice din România (RET), ce include 5 tipuri de IED-uri câte unul pentru fiecare tip de echipament electric primar: IED1 pentru întreruptor; IED2 pentru separator; IED3 pentru descărcător; IED4 pentru transformator de măsură de curent/tensiune; IED5 pentru transformator de putere/bobină de compensare.

Pentru implementarea practică a modelului matematic al reţelelor Bayesiene, am făcut calculele pe substaţia de 110kV care are în componenţă următoarele IED-uri:

- 15 IED-uri de tip IED1 (notate cu A1);
- 44 IED-uri de tip IED2 (notate cu A2);
- 7 IED-uri de tip IED3 (notate cu A3);
- 15 IED-uri de tip IED4 (notate cu A4);

Pentru calculul probabilităţii de defectare, trebuie calculată pentru fiecare nod Ai (IEDi) probabilitatea de defectare a priori. Aceste probabilităţi se obţin, colectând date privind numărul de căderi sau defectări din timpul testării IED-urilor.

Tabel 7.3.1. Număr de defecte/căderi pe tipuri de IED-uri pentru fiecare cabină de relee (CR)

Tip IED	**CR1**	**CR2**	**CR3**	**CR4**	**CR5**
IED1	2	0	4	0	0
IED2	1	3	12	0	3
IED3	0	1	3	0	0
IED4	0	0	4	0	1

Numărul de defecte prezentat în tabelul 7.3.1. reprezintă informaţii colectate pe o perioadă de 5 luni de teste cu IED-urile conectate la echipamentele electrice monitorizate, în staţia electrică.

S-au calculat probabilităţile de defectare pentru fiecare grup de IED-uri din cabinele de relee, unde acestea sunt conectate la câte un server local. În total sunt 5 astfel de servere locale.

De exemplu, pentru cabina de relee 1, se calculează probabilitatea de defectare a subsistemului de monitorizare şi control aferent cabinei de relee 1: **Server local CR1.**

În total sunt 11 IED-uri în cabina de relee 1: 2 IED-uri de tip IED1 (A1), 5 IED-uri de tip IED2 (A2), 2 IED-uri de tip IED3 (A3) şi 2 IED-uri de tip IED4 (A4). Utilizând aceste informaţii, se calculează probabilităţile pentru nodurile de tip IED din cabina de relee 1:
P(A1) = 2/11 = 0,18; P(A2) = 5/11 = 0,46; P(A3)= 2/11 = 0,18; P(A4) = 2/11 = 0,18.

Utilizând informaţiile privind numărul de căderi pentru fiecare IED, în perioada de 5 luni de test în staţia electrică se calculează probabilităţile a priori de defectare ale IED-urilor din cabina de relee 1:
P(E=defect|A1) = 0,2; P(E=defect|A2) = 0,04; P(E=defect|A3) = 0; P(E=defect|A4) = 0.

Se calculează probabilitatea de defectare a nodului tip server local din cabina de relee 1, notat în continuare CR1:
P(CR1) = P(A1) x P(E=defect|A1) + P(A2) x P(E=defect|A2) + P(A3) x P(E=defect|A3) + P(A4) x P(E=defect|A4) = 0,0544

Deci, probabilitatea de defectare a subsistemului de monitorizare din cabina de relee 1 este de 5,44%.

Pentru celelalte subsisteme de monitorizare din cabinele de relee CR2, CR3, CR4 respectiv CR5, se obţin următoarele rezultate:
P(**CR2**) este **4,16%**; P(**CR3**) este **20%**; P(**CR4**) este **0%**; P(**CR5**) este **5,33%**.

Pentru calculul probabilităţii de defectare a nodului de tip server central, notat în continuare SS1, se folosesc probabilităţile de defectare ale serverelor locale (subsistemele de monitorizare şi control) CR1, CR2, CR3, CR4 şi CR5, calculate anterior.

Modelând sistemul de monitorizare şi control din studiul de caz utilizând informaţiile din fişierul de configurare a staţiei electrice scris în limbajul SCL, prin cele 5 reţele Bayesiene, se obţine pentru acest sistem o **probabilitate de defectare de 6,98%.**

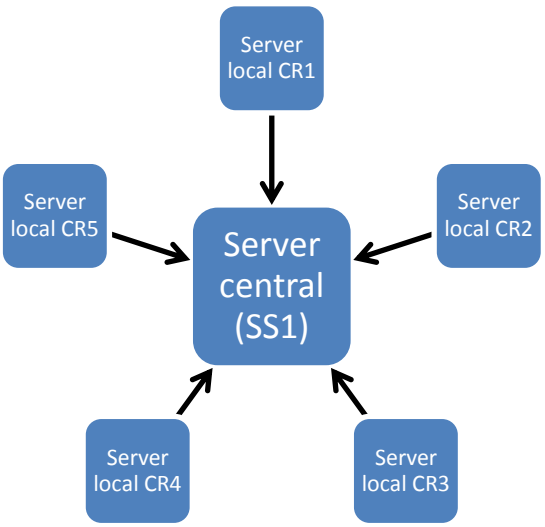

Fig. 7.3.1. Reţeaua Bayesiană pentru serverul central SS1 şi serverele locale

Vom nota cu S (sistem de monitorizare), nodul fiu în noua reţea Bayesiană. Acesta este influenţat de nodul SS1 (serverul central de BD) a cărei probabilitate de defectare am calculat-o anterior, de două noduri de tip server IED5 (Trafomon Trafo2 și BC – pentru monitorizarea transformatorului de putere respectiv bobinei de compensare) precum şi de nodul ce reprezintă calculatorul din camera de comandă pe care rulează aplicaţia tip client (nodul HMI - Human Machine Interface).

În formula lui Bayes de calcul a probabilităţii de defectare a nodului S, sunt folosite următoarele informaţii:

- Probabilitatea de defectare a priori pentru nodul SS1, P(E=defect|SS1) = 0,0895;
- Probabilitatea de defectare a priori pentru nodul IED5 Trafomon-T2, notat P(E=defect|A5-T2) și pentru nodul IED5 Trafomon-BC, notat P(E=defect|A5-BC);

  Pentru nodul A5 (IED5) aferent Trafomon-T2 nu a existat nici o cădere în timpul celor 5 luni, P(E=defect|A5-T2) = 0.

  Similar pentru nodul A5 aferent Trafomon-BC, P(E=defect|A5-BC) = 0.
- Probabilitatea de defectare a priori pentru nodul HMI, notata P(E=defect|HMI);

  Pentru calculatorul client din camera de comandă, utilizând informaţiile din perioada de teste, se calculează probabilitatea a priori de defectare: P(E=defect|HMI) = 2/9 = 0,22.

Conform calculelor efectuate, **probabilitatea de defectare a sistemului de monitorizare şi control S** (modelat prin reţeaua Bayesiană ce include serverul central de baze de date SS1, cele două servere locale pentru IED-urile de tip IED5 şi calculatorul client din camera de comandă) **este de 25,71%.**

8. Concluzii, contribuţii proprii şi planuri de cercetare pentru viitor

8.1. Concluzii

În cadrul tezei de doctorat am identificat principalele probleme ce afectează fiabilitatea sistemelor de monitorizare şi control a staţiilor electrice şi am propus soluţii de asigurare a calităţii acestor sisteme.

- În capitolul 2 am făcut o analiză a aspectelor de calitate software plecând de la un punct de vedere general, particularizând apoi pentru un sistem software de monitorizare şi control a unei staţii electrice. Astfel, sunt prezentate soluţii generale de asigurare a calităţii software precum: tehnici şi metode de prevenire a injectării defectelor, tehnici de eliminare a defectelor şi de izolare a acestora. S-a pus accentul pe activităţile de inspecţie şi testare, utilizarea blocurilor de recuperare, N-version programming, self-checking precum şi tehnici mai noi ca reconfigurarea şi reîntinerirea.
- În capitolul 3 au fost prezentate pe scurt câteva sisteme de monitorizare şi control, dezvoltate de cele mai reprezentative companii din domeniu, precum şi tehnologia Smart Grid. Au fost identificate problemele care pot apărea în dezvoltarea, funcţionarea şi operarea unui sistem de monitorizare şi control a unei staţii electrice, precum şi efectele acestor probleme.
- În capitolul 4 am propus soluţii concrete pentru asigurarea calităţii unui sistem de monitorizare şi control a unei staţii electrice, folosind ca studiu de caz un sistem dezvoltat şi instalat într-o staţie electrică de transformare din România. Este prezentată arhitectura şi funcţionalitatea sistemului, denumit EMCSIT (Echipament pentru Monitorizarea Complexă a Staţiilor de Înaltă Tensiune), la a cărui dezvoltare am participat. Sunt descrise soluţiile şi exemple de aplicare a acestor soluţii pentru asigurarea calităţii acestui sistem. Pentru îmbunătăţirea procesului de testare a aplicaţiilor (EMCSIT Server) ce achiziţionează date în timp real de la dispozitive tip IED (Intelligent Electronic Device), am dezvoltat un simulator software de IED-uri care generează pachete de date şi le trimite pe interfaţa serială a serverului la care sunt cuplate. Pachetele de date sunt generate ţinând cont de domeniile de valori ale mărimilor monitorizate de fiecare tip de IED, aplicând metoda *Pairwise testing*. În acest fel, se generează mult mai puţine cazuri de test faţă de numărul total al cazurilor care ar rezulta considerând toate combinaţiile posibile între

valorile de test selectate pentru mărimile monitorizate, însă se pot descoperi aproximativ 70% din erori în timpul testării.

- În capitolul 5 sunt prezentate aspecte generale privind standardul IEC61850 şi utilizarea acestuia în cadrul unei staţii electrice. Standardul prevede utilizarea de IED-uri pentru monitorizarea echipamentelor electrice primare. Pentru descrierea configuraţiei unei staţii electrice, în cadrul standardului este definit limbajul SCL. Structura generală a unui fişier SCL permite descrierea echipamentelor electrice din cadrul unei staţii precum şi asocierea IED-urilor destinate monitorizării şi controlului acestora.
- În capitolul 6 am propus utilizarea unor modele matematice pentru estimarea fiabilităţii sistemelor de monitorizare şi control a staţiilor electrice:
  - modelul de distribuţie Rayleigh, prin care am făcut aprecieri asupra eficienţei procesului de testare a aplicaţiilor care achiziţionează date în timp real de la dispozitivele de monitorizare (IED) a echipamentelor electrice;
  - modelul matematic al lanţurilor Markov, prin care se poate estima starea curentă a sistemului şi timpul de trecere dintr-o stare în alta. Se poate prezice momentul când sistemul va cădea;
  - reţelele Bayesiene, cu ajutorul cărora se poate calcula probabilitatea de defectare a sistemului de monitorizare şi control.
- Capitolul 7 descrie aplicaţia software pe care am dezvoltat-o pentru modelarea matematică a unui sistem de monitorizare şi control, în scopul estimării fiabilităţii sale. Aplicaţia conţine modulul *Parser SCL* şi modulul *Calcul BN*. Modulul *Parser SCL* citeşte informaţiile din fişierul scris în limbajul SCL şi pune la dispoziţia modulului de modelare a reţelei Bayesiene (*Calcul BN*), informaţii reprezentând IED-urile instalate în staţie. Aceste IED-uri monitorizează echipamentele electrice primare şi sunt componente ale sistemului de monitorizare şi control a staţiei electrice. În implementarea modelului matematic al reţelelor Bayesiene din modulul *Calcul BN*, aceste IED-uri sunt reprezentate ca fiind nodurile reţelei Bayesiene. Legăturile între noduri sunt editabile, existând posibilitatea definirii celor două tipuri de noduri: noduri de tip fiu şi noduri de tip părinte. Configuraţia staţiei electrice în limbajul SCL conform standardului IEC61850 poate fi completată, adăugând noduri de tip server şi noduri pe care rulează aplicaţii software. Utilizând informaţii privind rata de defectare a IED-urilor din studiul de caz (sistemul EMCSIT), am calculat probabilitatea de defectare a serverelor locale, a serverului central precum şi a întregului sistem de monitorizare şi control a staţiei electrice. Scopul acestor

calcule este estimarea fiabilităţii sistemului pentru a putea interveni prompt şi eficient în cadrul componentelor sau subsistemelor care prezintă o probabilitate de defectare ridicată.

8.2. Contribuţii proprii

În continuare, menţionez principalele contribuţii originale în domeniul tezei de doctorat, care au fost prezentate în diferite capitole ale tezei:

- Am analizat problemele care pot cauza căderi ale unui sistem de monitorizare şi control a unei staţii electrice şi am propus o serie de soluţii de asigurare a calităţii unui astfel de sistem, care pot contribui la creşterea fiabilităţii sale (capitolele 2 şi 3). Aceste soluţii presupun:
    - Utilizarea unor metode eficiente de testare a aplicaţiilor din componenţa sistemului.

      Analiza prezentată în paragraful 6.1, bazată pe date reale culese în timpul procesului de testare, a evidenţiat importanţa dezvoltării unor simulatoare care să permită testarea înainte de instalarea în mediul real de funcţionare, a aplicaţiilor care primesc date în timp real de la dispozitivele de tip IED, cuplate la diferite tipuri de echipamente electrice. Astfel de teste permit descoperirea unui număr însemnat de defecte în aceste aplicaţii, înainte de testarea întregului sistem de monitorizare în mediul real de funcţionare (instalat în staţie). Efectul financiar poate fi semnificativ: timpul de testare cu întregul sistem de monitorizare instalat în staţia electrică (în funcţiune) este limitat şi din acest motiv şi descoperirea defectelor în această perioadă este redusă.
    - Asigurarea securităţii serverelor, a aplicaţiilor software din componenţa sistemului de monitorizare şi control şi a transmisiei datelor între diversele componente ale sistemului.
    - Implementarea unor tehnici de toleranţă la defecte: folosirea blocurilor de recuperare, tehnici de duplicare, tehnici de reconfigurare şi reîntinerire.
- Pornind de la analiza efectuată în capitolele 2 si 3, am dezvoltat un simulator de IED-uri care poate fi configurat pentru diferite tipuri de echipamente electrice (cap. 4). Acesta permite testarea automată a aplicaţiilor (denumite aplicaţii server în sistemul studiu de caz) care preiau datele transmise în timp real de dispozitivele tip IED incluse într-un sistem de monitorizare şi control a unei staţii electrice. În plus, simulatorul permite efectuarea de teste pentru cazurile în care mai multe IED-uri sunt cuplate la aceeaşi

aplicaţie, utilizând protocolul Daisy-Chain. Astfel de teste sunt extrem de dificil de realizat în absenţa unui astfel de simulator, deoarece nu sunt disponibile fizic, în timp util, un număr suficient de mare de IED-uri pentru conectarea la aplicaţiile server. Pachetele de date transmise de simulator au fost generate prin metoda *Pairwise testing*, o metodă de testare „black-box" foarte eficientă, care poate conduce la descoperirea a aproximativ 70% din defectele existente într-un software, prin utilizarea unui număr relativ redus de cazuri de test.

- Am analizat, folosind modelul Rayleigh, eficienţa eliminării defectelor pe parcursul dezvoltării şi testării unor componente dintr-un sistem de monitorizare şi control a unei staţii electrice (paragraful 6.1.). Analiza a evidenţiat importanţa eliminării unui număr cât mai mare de defecte înainte de testarea în condiţii reale de exploatare.
- Am propus modelarea unui sistem de monitorizare şi control printr-un lanţ Markov care presupune existenţa a 4 stări ale sistemului: stare bună, stare acceptabilă, stare proastă şi stare inacceptabilă. S-a propus o metodă de a estima starea curentă a sistemului prin încadrarea într-una din cele 4 stări propuse, utilizând drept variabilă aleatoare: „numărul de pachete de date ratate la intervalul fixat de achiziţie de către sistem". Este descrisă metoda de calcul a FPT (first passage times) – timpii de tranziţie între stări – care ajută la predicţia momentului de timp când un sistem poate ajunge într-o stare viitoare [Urs8-20]. Modelul propus, cu 4 stări ale sistemului permite predicţia momentului când sistemul poate ajunge într-o stare anterioară celei inacceptabile sau chiar în starea inacceptabilă, astfel încât să se poată interveni cât mai rapid pentru remedierea defectelor. Implementarea acestui model poate preveni situaţii grave generate de funcţionarea inacceptabilă sau chiar nefuncţionarea sistemului.
- În scopul estimării probabilităţii de apariţie a defectelor în timpul operării sistemului, am propus utilizarea unei reţele Bayesiene (paragraful 6.3.). Am propus modelarea sistemelor de monitorizare şi control definite conform standardului IEC61850, printr-o astfel de reţea (paragraful 6.3.2.). De asemenea, am modelat printr-o reţea Bayesiană sistemul de monitorizare şi control prezentat ca studiu de caz în capitolul 4 (paragraful 6.3.3) [Urs8-6].
- Am descris configuraţia unei staţii electrice reale, conform standardului IEC61850 (limbajul SCL). Această configuraţie include echipamentele electrice primare din staţie şi IED-urile asociate ce alcătuiesc sistemul de monitorizare şi control a staţiei electrice.
- Am dezvoltat o aplicaţie software (capitolul 7) care:

- Preia dintr-un fişier în format SCL configuraţia unui sistem de monitorizare şi control a unei staţii electrice.
- Permite adăugarea de noi noduri definite de utilizator: noduri de tip calculator Server local, noduri de tip calculator Client, noduri de tip Server de baze de date, etc.
- Generează reţeaua Bayesiană prin definirea nodurilor de tipul părinte şi a nodurilor de tip fiu utilizând date din configuraţia sistemului de monitorizare şi control;
- Permite adăugarea de informaţii pentru nodurile de tip părinte: numărul de defecte/căderi pe o anumită perioadă de timp sau probabilitatea a priori de defectare a nodului;
- Estimează probabilitatea de defectare a sistemului şi probabilitatea a posteriori de defectare a fiecărui nod al reţelei;

▪ În cap. 7.3. sunt redate sintetic rezultatele obţinute pentru sistemul studiu de caz cu ajutorul aplicaţiei.

8.3. Planuri de cercetare pentru viitor

În viitor voi încerca să studiez şi să aprofundez următoarele aspecte:

- Utilizarea practică a metodei N-version programming în dezvoltarea ulterioară a unor astfel de sisteme software.
- Simulatorul de IED-uri poate fi dezvoltat în continuare pentru a fi compatibil şi cu alte protocoale de comunicaţie şi tipuri de IED-uri.
- Implementarea posibilităţii de vizualizare grafică a unei reţele Bayesiene.
- Completarea aplicaţiei de calcul a fiabilităţii sistemului cu un modul de calcul a unor metrici software relevante, pentru componentele software ale sistemului.
- Completarea modelului reţelei Bayesiene prin includerea în calcule a datelor pentru echipamente electrice care nu sunt monitorizate, deci nu au IED ataşat dar prin defectarea lor mecanică/electrică pot afecta funcţionarea sistemului de monitorizare şi control a staţiei electrice.
- Implementarea modelului matematic pentru predicţia timpilor de defectare bazat pe lanţul Markov, propus în paragraful 6.2.

Lista lucrărilor autorului (selecţie)

[Urs8-1] **V. Ursianu**, E. Ursianu, R. Ursianu, "Regression model approach through proper roots", Scientific Bulletin UPB. Applied Mathematics and Physics Vol. 72/2010, Iss.4/33.

[Urs8-2] **V. Ursianu**, M. Iliescu, "Statistică Aplicată în Inginerie - aspecte teoretice", Ed. Bren.

[Urs8-3] C. Moldoveanu, V. Brezoianu, A. Vasile, **V. Ursianu**, F. Goni, C. Radu, I. Ioniţa, M. Avramescu, S. Zaharescu, B. Toader: "Intelligent electronic system for continuous monitoring and diagnostic of high voltage substations", CMDM2011 International Conference on Condition Monitoring, Diagnosis and Maintenance, Bucureşti.

[Urs8-5] **V. Ursianu**, Fl. Moldoveanu, R. Ursianu, E. Ursianu: "Software quality assurance for monitoring and control systems in the energy field", CSCS18 International Conference on Control Systems and Computer Science, 2011, Bucureşti.

[Urs8-6] **V. Ursianu**, R. Ursianu, E. Ursianu: "Bayesian Networks to predict Software Quality", SPSR14 Conferinţa Societăţii de Probabilităţi şi Statistică, 2011, Bucureşti.

[Urs8-8] C. Moldoveanu, V. Brezoianu, A. Vasile, **V. Ursianu**, F. Goni, C. Radu, I. Ioniţă: "Intelligent System for the On-Line Real Time Monitoring of High Voltage Substations" IEEE ISGT2010 Innovative Smart Grid Conference, Goteborg (cotată ISI).

[Urs8-11] C. Moldoveanu, **V. Ursianu**, V. Brezoianu, A. Vasile, I. Ioniţa, S. Gal, C. Diaconu, V. Zaharescu, T. Fagarasan, M. Oltean, G. Moraru: "Solutions for life management and maintenance optimization for large power transformers", CMD International Conference on Condition Monitoring and Diagnosis 2010, Tokyo (cotată ISI).

[Urs8-17] **V. Ursianu**, R. Ursianu, E. Ursianu: "Models for Determining the Quality of an Electrical Equipment" , CSCS17 International Conference on Control Systems and Computer Science 2009, Bucureşti.

[Urs8-18] **V. Ursianu**, R. Ursianu, E. Ursianu: "Mathematical Model for Deterioration Process of an Electrical Station with Multiple Components", MACMESE 2008, WSEAS Conference, Bucureşti (cotată ISI).

[Urs8-20] C. Moldoveanu, R. Ursianu, E. Ursianu, E. Mihalcea, M. Nestor, F. Goni, L. Goia, P. Curiac, **V. Ursianu**: "Determine the optimal moments for investigating technical state of primary equipments for the purpose of assuring the safety levels imputed by the National Company Transelectrica – România", CMD2008 International Conference on Condition Monitoring and Diagnosis, Beijing (cotată ISI).

Bibliografie selectivă

[Swq9-1] “Software Quality Engineering”, J. Tian.

[Swq9-5] “Software testing and Quality assurance”, K. Naik, P. Tripathy. Ed. Wiley.

[Swq9-6] ”Curs avansat de Ingineria Programelor – Asigurarea calităţii software”, Fl. Moldoveanu.

[Swq9-7] “Methodology of N-version programming”, A. Avizienis, Ed. Wiley.

[Swq9-8] “Metrics and Models în Software Quality Engineering”, S. Khan, Ed. Wiley.

[Swq9-9] “Recovery blocks in action: A system supporting high reliability”, T. Anderson, R. Kerr.

[Swq9-13] Curs Sisteme distribuite – Toleranta la defecte, D. Petcu.

[Smg9-1] “NOVA EMCSIT – Sistem complex de monitorizare on-line a unei staţii electrice”, C. Moldoveanu, V. Ursianu, A. Vasile, V. Brezoianu.

[Smg9-2] “Sisteme de timp-real”, T. Letea, Ed. Albastră.

[Smg9-9] The National Institute of Standards and Technology (NIST) Smart Grid Conceptual model http://smartgrid.ieee.org/ieee-smart-grid/smart-grid-conceptual-model

[Net9-1] IPv6 website http://ro.wikipedia.org/wiki/IPv6

[Net9-2] ArenaIT

http://www.arenait.net/2007/03/12/tot-ce-trebuie-sa-stiti-despre-ipv6.html

[Iec9-3] “IEC 61850 Power Industry Communications Standard”, B. Lydon.

[Iec9-9] “IEC 61850 Object Model and Configuration Language”, C. Brunner.

[Iec9-11] Visual SCL, Applied Systems Engineering website

http://www.ase-systems.com/iec-61850/visual-scl.asp

[Swm9-3] Microsoft MSDN website http://blogs.msdn.com/b/codeanalysis

[Swm9-4] “Software Metrics”, Fl. Moldoveanu.

[Ban9-4] “Event-based Failure Prediction. An Extended Hidden Markov Model Approach”, F. Salfner. PhD Dissertation Thessis.

[Ban9-5] Math 115, Calculus II with Probability&Metrics, J. Guffin.

http://www.math.upenn.edu/~guffin/teaching/spring11/index.html

[Ban9-7] “Teoria deciziilor statistică”, V. Preda. Ed. Academiei Romane.

[Ban9-12] “Dicţionar explicativ de statistică”, V. Clocotici.

[Rel9-1] “Fiabilitatea în arhitectură calculatoarelor”, M. Budiu.