



KAUNO TECHNOLOGIJOS UNIVERSITETAS
FUNDAMENTALIŲJŲ MOKSLŲ FAKULTETAS
TAIKOMOSIOS MATEMATIKOS KATEDRA

Tadas Eimontas

TIKIMYBINĖS DINAMIKOS
MODELIAVIMAS IR PATIKIMUMO
ANALIZĖ

Magistro darbas

Vadovas
dr. R. Alzbutas

KAUNAS, 2007



KAUNO TECHNOLOGIJOS UNIVERSITETAS
FUNDAMENTALIŲJŲ MOKSLŲ FAKULTETAS
TAIKOMOSIOS MATEMATIKOS KATEDRA

TVIRTINU
Katedros vedėjas
prof. dr. J.Rimas
2007 06 06

TIKIMYBINĖS DINAMIKOS
MODELIAVIMAS IR PATIKIMUMO
ANALIZĖ

Taikomosios matematikos magistro baigiamasis darbas

Vadovas
dr. R. Alzbutas
2007 05 31

Recenzentas

habil. dr. J. Augutis
2007 05 31

Atliko
FMMM 5/1 gr. stud.
T. Eimontas
2007 05 31

KAUNAS, 2007

KVALIFIKACINĖ KOMISIJA

Pirmininkas: Leonas Saulis, profesorius (VGTU)

Sekretorius: Eimutis Valakevičius, docentas (KTU)

Nariai: Algimantas Jonas Aksomaitis, profesorius (KTU),

Arūnas Barauskas, dr., UAB „Elsis“ generalinio direktoriaus pavaduotojas,

Vytautas Janilionis, docentas (KTU),

Zenonas Navickas, profesorius (KTU),

Vidmantas Povilas Pekarskas, profesorius (KTU),

Rimantas Rudzkis, habil.dr., banko „DnB NORD“ vyriausiasis analitikas.

Eimontas T. Simulation and reliability analysis of probabilistic dynamics: Master's work in applied mathematics / supervisor dr. R. Alzbutas; Department of Applied mathematics, Faculty of Fundamental Sciences, Kaunas University of Technology. – Kaunas, 2007. – 82 p.

SUMMARY

The current probabilistic safety analysis is not capable of estimating the reliability of the complex dynamic systems where the interactions occur between hardware, software and human actions. In the safety analysis of these systems the time factor is as much important as it joins an evolution of physical variables and stochastic events. In this master thesis the simulation and reliability analysis of the probabilistic dynamics are considered. The new approach of stimulus based probabilistic dynamics is used for the Monte Carlo simulations of the dynamic system. The developed methodology was applied for the safety analysis of the loss of the coolant accident in the nuclear reactor. Besides the assessment of the probability of system failure the scenario analysis was accomplished. The essential events were identified. The uncertainty and sensitivity analysis revealed that the failure probability had a wide range of the distribution due to the uncertainty of twelve simulation parameters. Four main parameters were identified as their uncertainty had the biggest correlation with the uncertainty of the system failure. For the complete reliability analysis the relations between the failure probability and the system characteristics were determined.

TURINYS

TURINYS	5
PAVEIKSLŲ SĄRAŠAS	6
LENTELIŲ SĄRAŠAS	7
ĮVADAS	8
1. BENDROJI DALIS	10
1.1. TIKIMYBINĖ SAUGOS ANALIZĖ	10
1.1.1. Rizikos įvertinimo problematika.....	10
1.1.2. Gedimų ir įvykių medžiai.....	12
1.1.3. Naujų analizės metodų plėtojimas.....	14
1.1.4. Tikimybinės dinamikos teorija	15
1.1.5. Stimuliuojamos dinamikos teorija.....	17
1.2. NEAPIBRĖŽTUMO ANALIZĖS METODOLOGIJA.....	22
1.2.1. Neapibrėžtumo analizės tikslai ir taikymas.....	22
1.2.2. Neapibrėžtumų įtaka modeliavimo rezultatams	24
1.2.3. Modelio neapibrėžtumo įvertinimas.....	26
1.2.4. Jautrumo analizės taikymai.....	27
1.2.5. Jautrumo analizės metodai.....	28
2. TIRIAMOJI DALIS	31
2.1. TIKIMYBINĖS DINAMIKOS PATIKIMUMO ANALIZĖ	31
2.1.1. Tikimybinės dinamikos sistema	31
2.1.2. Sistemos formalizavimas	35
2.1.3. Imitacinio modeliavimo metodologija.....	40
2.1.4. Sistemos tikimybinė saugos analizė.....	43
2.2. NEAPIBRĖŽTUMO IR JAUTRUMO ANALIZĖ.....	48
2.2.1. Duomenų ir sistemos modelio neapibrėžtumai	48
2.2.2. Modelio neapibrėžtumo analizė	51
2.2.3. Modelio jautrumo analizė	54
IŠVADOS IR REZULTATAI	58
PADĖKOS	59
LITERATŪRA	60
1 PRIEDAS. PROGRAMINĖ ĮRANGA	62

PAVEIKSLŲ SĄRAŠAS

1.1 PAV. RIZIKOS VERTINIMO SCHEMA	10
1.2 PAV. AUTOMOBILIŲ SUSIDŪRIMO SANKRYŽOJE GEDIMŲ MEDIS	12
1.3 PAV. PRIEŠGAISRINĖS SAUGOS ĮVYKIŲ MEDIS	13
1.4 PAV. DALINAI DETERMINISTINIS SISTEMOS KITIMAS	15
1.5 PAV. SISTEMOS KITIMAS DINAMIKOJE I ĮVYKUS STIMULO F AKTYVACIJAI	18
1.6 PAV. IŠLIEKANTYS UŽDELSIMAI PO PERĖJIMO Į NAUJĄ DINAMIKĄ	20
1.7 PAV. GALIMOS SU STIMULAIS SUSIJUSIOS SITUACIJOS	21
1.8 PAV. SISTEMOS MODELIO REZULTATŲ NEAPIBRĖŽTUMĄ ĮTAKOJANTI SEKA	24
1.9 PAV. MODELIO PARAMETRŲ NEAPIBRĖŽTUMO RYŠYS SU REZULTATŲ NEAPIBRĖŽTUMU	26
1.10 PAV. SKLAIDOS GRAFIKŲ PAVYZDŽIAI	28
2.1 PAV. ŠILUMNEŠIO PRARADIMO AVARIJOS POVEIKIS	31
2.2 PAV. VIDINIAI MODELIO SĄRYŠIAI TARP TIKIMYBINIŲ IR DETERMINISTINIŲ CHARAKTERISTIKŲ ..	40
2.3 PAV. DINAMIKŲ PERĖJIMAI	43
2.4 PAV. VANDENILIO MOLIŲ KIEKIO KITIMO GRAFIKAS	44
2.5 PAV. VANDENS GARŲ MOLIŲ KIEKIO KITIMO GRAFIKAS	44
2.6 PAV. SLĖGIO KITIMO GRAFIKAS	44
2.7 PAV. APSAUGINIO APGAUBO PAŽEIDIMO DAŽNIO KITIMAS LAIKE	45
2.8 PAV. APGAUBO PAŽEIDIMO DAŽNIŲ HISTOGRAMA	52
2.9 PAV. APGAUBO PAŽEIDIMO DAŽNIO KITIMO LAIKE FUNKCIJOS	53
2.10 PAV. APGAUBO PAŽEIDIMO DAŽNIO EMPIRINĖS CHARAKTERISTIKOS	53
2.11 PAV. PIRSONO KORELIACIJOS KOEFICIENTAI TARP MODELIO PARAMETRŲ IR PAŽEIDIMO DAŽNIO	55
2.12 PAV. DALINĖS KORELIACIJOS KOEFICIENTAI	56
2.13 PAV. STANDARTIZUOTI REGRESIJOS KOEFICIENTAI	57

LENTELIŲ SĄRAŠAS

1.1 LENTELĖ MINIMALUS SKAIČIAVIMŲ KIEKIS (<i>A</i> , <i>B</i>) TOLERANCIJOS INTERVALUI.....	27
1.2 LENTELĖ MINIMALUS SKAIČIAVIMŲ KIEKIS (<i>A</i> , <i>B</i>) TOLERANCIJOS RIBAI.....	27
2.1 LENTELĖ SISTEMOS PARAMETRAI IR JŲ PRADINĖS REIKŠMĖS	32
2.2 LENTELĖ IŠSISKIRIANČIO VANDENILIO MASĖS SRAUTAS	32
2.3 LENTELĖ VANDENS GARŲ KONCENTRACIJOS KITIMAS ĮJUNGUS APGAUBO ATŠALDYMO SISTEMĄ	33
2.4 LENTELĖ TEMPERATŪROS KITIMAS ĮJUNGUS APGAUBO ATŠALDYMO SISTEMĄ.....	33
2.5 LENTELĖ SISTEMOS ĮVYKIAI	35
2.6 LENTELĖ SISTEMOS DINAMIKOS	36
2.7 LENTELĖ ĮVYKIŲ STIMULŲ IR UŽDELSIMŲ CHARAKTERISTIKOS	37
2.8 LENTELĖ SUMINIS VANDENILIO MASĖS SRAUTAS.....	37
2.9 LENTELĖ VANDENS GARŲ KIEKIO SKAIČIAVIMAS PRIKLAUSOMAI NUO ESAMOS DINAMIKOS	38
2.10 LENTELĖ BENDRAS IŠSISKIRIANČIO VANDENILIO SRAUTAS	39
2.11 LENTELĖ TEMPERATŪROS KITIMAS PRIKLAUSOMAI NUO ESAMOS DINAMIKOS	39
2.12 LENTELĖ SISTEMOS MODELIAVIMO REZULTATAI.....	45
2.13 LENTELĖ APSAUGOS SISTEMŲ SUVEIKIMO SCENARIJAI	46
2.14 LENTELĖ SCENARIJŲ ATSIKIRIMO TIKIMYBĖS	46
2.15 LENTELĖ SCENARIJŲ TIKIMYBINĖS CHARAKTERISTIKOS	47
2.16 LENTELĖ SISTEMOS KINTAMŲJŲ PRADINIŲ REIKŠMIŲ IŠSIBARSTYMO RIBOS.....	48
2.17 LENTELĖ DĖL PLIENO IR CIRKONIO OKSIDACIJOS IŠSISKIRIANČIO VANDENILIO MASĖS SRAUTO NEAPIBRĖŽTUMAI.....	49
2.18 LENTELĖ VANDENS GARŲ KIEKIO KITIMO NEAPIBRĖŽTUMAI	49
2.19 LENTELĖ PAŽEIDIMO DAŽNIO NEAPIBRĖŽTUMAS.....	50
2.20 LENTELĖ SISTEMOS DUOMENŲ IR PARAMETRŲ NEAPIBRĖŽTUMAI	50
2.21 LENTELĖ APGAUBO PAŽEIDIMO DAŽNIO EMPIRINĖS CHARAKTERISTIKOS.....	52

ĮVADAS

Dėl spartaus technologijų naudojimo paskutiniaisiais dešimtmečiais kuriama vis daugiau sudėtingų sistemų, kurių saugos užtikrinimui turi būti vertinamas techninės ir programinės įrangos patikimumas bei žmogaus-operatoriaus veiksmai. Analizuojant tokias sistemas ypatingai svarbią reikšmę turi laiko faktorius ir su juo susiję determinuoti fiziniai procesai bei stochastiniai įvykiai.

Šiuolaikinių saugos įvertinimo metodų ištakos siekia 1930-1940 m., tačiau šie metodai buvo apibendrinti tik apie 1980-uosius metus C.Starro, R.Rudmano, C.Whipple, W.D.Rowe's, J.H.Bickelo, D.L.Kelly'o, T.J.Leahy'o ir kitų autorių darbuose [4,22]. Tyrimai buvo pradėti struktūrinėje patikimumo analizėje ir sistemų analizėje, kuri yra susijusi su apkrovos/atsparos uždaviniais, aptinkamais dažniausiai pastatų ar statybų konstrukcijų patikimumo vertinimuose. 1961 m. H.Watson pirmą kartą patikimumo analizėje panaudojo gedimų medį, kuris vėliau tapo vienu populiariausių metodų, taikomų sistemų analizėje [5].

Šiuo metu tikimybinės analizės pagrindas vis dar yra gedimų bei įvykių medžiai. Jų pagrindu sukurtiems modeliams papildomai atliekama neapibrėžtumo ir jautrumo analizė [1,2,10]. Pastaruoju metu, taip pat yra plėtojama tikimybinės dinamikos teorija [8,15], kurios tikslas įvertinti sistemos patikimumą atsižvelgiant į laiko faktorių. Tarp plėtojamų teorijų išsiskiria stimuliuojamos dinamikos tyrimo metodologija [19]. Ji taikoma analizuojant sistemas, kuriose atsitiktiniai įvykiai atsiranda su tam tikru uždelsimu. Kadangi šios plėtojamos teorijos yra sąlyginai naujos, jos nėra pagrįstos realių sistemų modeliavimo rezultatais. Taip pat nėra tiksliai apibrėžtos jų taikymo bei sistemų modeliavimo metodikos.

Atsitiktiniai atsirandantys stimulai ir uždelsti stochastiniai procesai yra būdingi daugeliui dinaminių sistemų. Tokių sistemų patikimumo analizė yra problematiška, kadangi šiuo metu plačiai taikomi gedimų bei įvykių medžiai neįvertina laikinių charakteristikų. Be to, dinaminių sistemų duomenys retai kada būna išsamūs ir tinkami matematiniam modeliavimui. Yra žinoma, kad kartais vien dėl duomenų trūkumo ir neatsižvelgimo į esamą neapibrėžtumą yra daromi klaidingi sprendimai.

Darbo tikslas yra sukurti tikimybinės dinamikos modeliavimo metodiką išplėtojant susijusias patikimumo analizės priemones bei jas pritaikyti realios sistemos tyrimui. Tiriamojo darbo tematika siejasi su šiomis svarbiomis mokslinių tyrimų sritimis:

- saugi sudėtingų techninių dinaminių sistemų eksploatacija;
- sudėtingų uždelsto veikimo sistemų modeliavimo efektyvumo didinimas;
- techninių sprendimų pagrindimas ir optimizavimas atsižvelgiant į neapibrėžtumą.

Pagrindiniai darbo uždaviniai: sukurti imitacinio modeliavimo metodologiją ir ją pritaikyti tiriant realią sistemą, atlikti išsamią jos patikimumo analizę, įvertinant ne tik saugos sistemų pažeidimo ar sugedimo tikimybę, bet taip pat ir gauto įverčio neapibrėžtumą; nustatyti nagrinėjamos sistemos

pažeidimą labiausiai įtakojančius faktorius bei pažeidimo tikimybės neapibrėžtumą lemiančius modelio parametrus.

Sprendžiant užsibrėžtus uždavinius, darbe buvo pritaikyta pažangi stimuliuojamos dinamikos teorija, šiuo metu neturinti plataus praktinių modelių pagrindimo. Pritaikyta imitacinio modeliavimo metodika suteikė galimybę atlikti visapusišką šilumnešio praradimo avarijos saugos analizę.

Reikšmingiausi darbo rezultatai yra susiję su neapibrėžtų įvykių ir dinaminių sistemų saugos analize, siekiant padidinti jų patikimumą. Didžioji atlikto darbo taikymų dalis skirta techninėms dinaminėms sistemoms. Darbe išnagrinėti ir išplėtoti modeliavimo metodai, kurie yra tinkami tirti sistemų patikimumą, susijusį su uždelstais pavojingais įvykiais ar rizikingais operatorių sprendimais.

1. BENDROJI DALIS

1.1. TIKIMYBINĖ SAUGOS ANALIZĖ

1.1.1. RIZIKOS ĮVERTINIMO PROBLEMATIKA

Daugelis XX-o amžiaus katastrofomis vadinamų avarių branduolinės energetikos, chemijos, transporto, naftos ir kitose pramonės šakose atnešė didžiulių nuostolių ir aukų. Įvykusios avarijos ypač išryškino rizikos valdymo ir įvertinimo svarbą tiek normalios eksploatacijos, tiek ir avarių metu. Rizikos įvertinimo metodai pradėti taikyti dar septintajame dešimtmetyje, o 1980-1990 metais stebimas ženklus šuolis tiek metodų išvystymo, tiek ir taikymų srityse. Taigi rizikos įvertinimo problematika tapo atskira mokslo šaka, vadinama rizikos analize arba rizikos įvertinimu, kuri susieja tikimybių teoriją, matematinę statistiką ir technologijos mokslus [2].

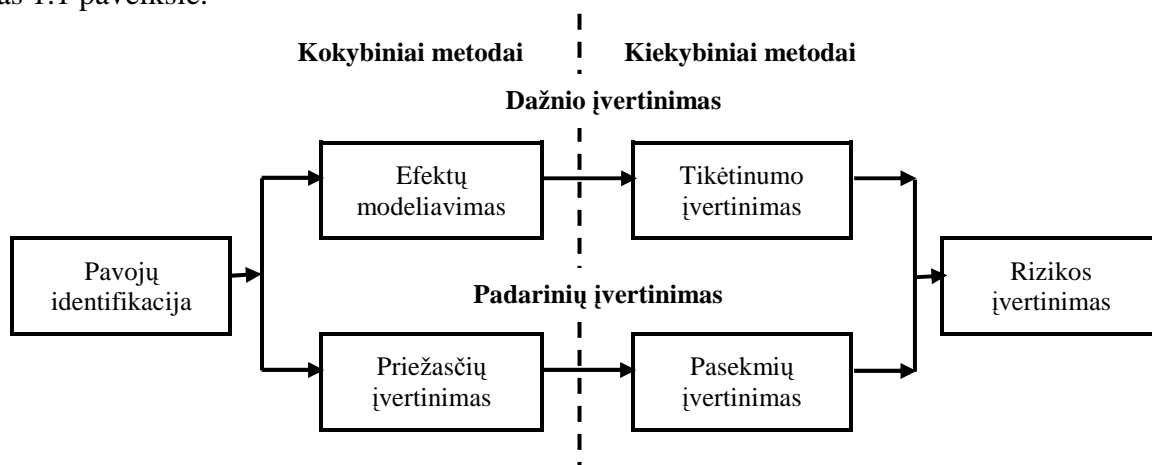
Rizika gali būti vertinama tiek kiekybiškai, tiek kokybiškai. Kokybine prasme, kiekvienas rizikos šaltinis nesant arba neveikiant apsaugos barjerams sudaro galimybę įvykti avarijai, gyvybės praradimui arba kitiems nuostoliams. Ši galimybė vadinama rizika.

Kiekybine prasme, fundamentalūs rizikos analizės klausimai yra šie:

1. Kas turi atsitikti, kad įvyktų pavojingas įvykis?
2. Kaip dažnai tai gali atsitikti?
3. Kokios pasekmės yra numatomos, įvykiui atsitikus?

Atsakant į pirmą klausimą, nustatomi įvykiai ar jų grupės (scenarijai), sąlygojantys pavojingą įvykį. Toliau, pagal antrą klausimą, įvertinamos šių scenarijų tikimybės arba dažniai ir galimos pasekmės (trečias klausimas).

Rizikos analizė procesas aprėpiantis kokybinį ir kiekybinį rizikos vertinimą schematiškai pavaizduotas 1.1 paveiksle.



1.1 pav. Rizikos vertinimo schema

Vienas dažniausių literatūroje pateikiamų rizikos apibrėžimų yra šių trinarių aibė [21]:

$$R = \langle S_i, P_i, C_i \rangle, \quad i = 1, 2, \dots, n, \quad (1.1)$$

kur S_i – įvykių scenarijus, sukeliantis pavojų,

P_i – scenarijaus i tikimybė,

C_i – scenarijaus i pasekmė, (pvz., nuostolių kiekybinis įvertinimas),

n – scenarijų skaičius.

Scenarijai parodo, kas gali atsitikti su nagrinėjamu objektu, o jų tikimybės įvertina, kaip dažnai gali toks scenarijus įvykti iš tikrųjų. Scenarijų pasekmės gali būti išreiškiamos labai įvairiai, pvz., žuvusių žmonių skaičiumi, nuostoliais, kurie gali būti įvertinti pinigais, aplinkos užterštumu, įmonės prestižu ir pan. Įvertinant konkrečių reiškinių riziką stengiamasi visų scenarijų pasekmes išreikšti vienu matu, pvz., draudimo versle visų nelaimingų atsitikimų nuostoliai išreiškiami pinigais, avarijų pasekmės atominėse elektrinėse – kiekiu radioaktyvių medžiagų, patenkančių į aplinką ir t. t. Jeigu šis matas yra skaitinis, tai atsiranda galimybė įvertinti riziką vien kiekybiškai, kaip nepalankaus įvykio tikimybės arba dažnio q sandaugą iš nuostolių dydžio A , t. y. rizika nusakoma šia išraiška:

$$R = Aq \quad (1.2)$$

Detalesnė rizikos įvertinimo analizė rodo, kad nors ir egzistuoja bendri rizikos įvertinimo metodologiniai faktoriai, metodų taikymas labai skirtingose ir dažnai mažai susijusiose srityse yra komplikuoatas dėl skirtingų analizės tikslų, terminų ir techninių ypatumų.

Vienas iš techninių sistemų rizikos įvertinimo uždavinių tipų yra taip vadinamas apkrovos/atsparos uždavinys. Šiuose uždaviniuose sistemos apkrova atskiriama nuo sistemos atsparos atlaikyti duotą apkrovą. Šie uždaviniai dažnai vadinami struktūrinės saugos uždaviniais ir dažnai sutinkami statybinių konstrukcijų patikimumo analizėje, kur ekstremalūs gamtos reiškiniai (pvz., žemės drebėjimai, uraganai, potvyniai) vadinami apkrova, o statybinių objektų atspara turi viršyti apkrovą. Apkrova ir atspara yra nepriklausomi, tačiau kiekvienas iš jų gali būti sudarytas iš keleto veiksmų.

Kita rizikos įvertinimo uždavinių grupė yra susijusi su techninėmis sistemomis, sudarytomis iš daugelio elementų, kurių kiekvieno patikimumo charakteristikos gali būti apskaičiuotos. Tokių sistemų pavyzdžiais galėtų būti branduolinio reaktoriaus aušinimo, elektros tiekimo, procesų valdymo, mechaninės ar kitokios sistemos.

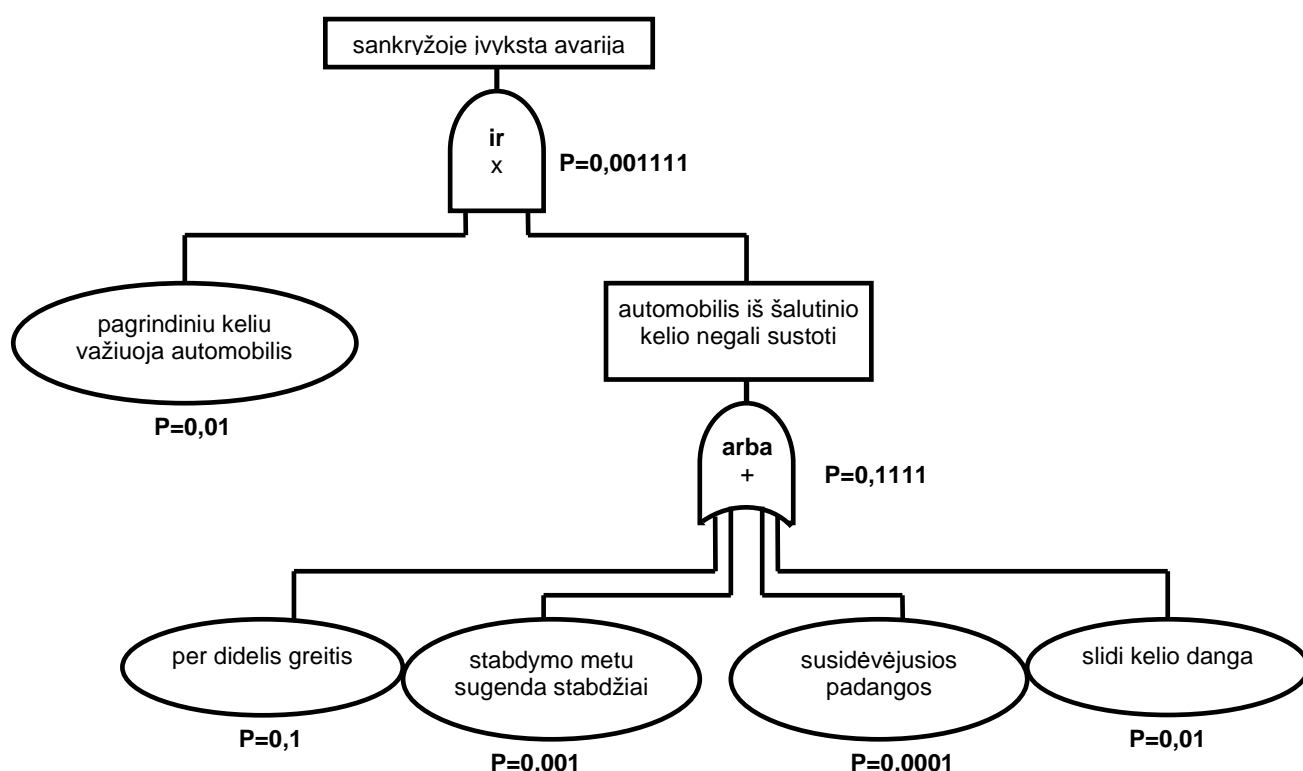
Sistemų rizikos analizė yra pagrįsta Markovo modeliais, kurių prigimtis atitinka techninės sistemos elgesį, t. y. ateities gedimai nepriklauso nuo praeities. Daugelio techninių sistemų darbas visiškai atitinka šią Markovo modelių prielaidą.

Sudėtingų techninių sistemų rizikai įvertinti dažniausiai praktikoje taikomas metodas yra tikimybinė saugos analizė, pagrįsta gedimų ir įvykių medžių analize. Šis metodas gali būti analizuojamas kaip tam tikras Markovo modelių atvejis.

1.1.2. GEDIMŲ IR ĮVYKIŲ MEDŽIAI

Vienas labiausiai išvystytų ir taikomų metodų techninių sistemų patikimumo uždaviniuose yra įvykių/gedimų medžių analizė (atskiras Markovo modelių atvejais). Įvykių/gedimų medžių metodas loginiu pagrindu apjungia pradinius gedimus sistemoje (inicijuojančius įvykius) su įvairių scenarijų pasekmėmis (pvz., avarijomis). Metodo pagrindą sudaro gedimų medžiai, kurie nurodo loginius ryšius tarp sistemos elementų (bazinių įvykių) būsenų ir sistemos būsenos (viršutinio įvykio). Šis metodas buvo sukurtas Bell laboratorijoje 1962 metais [5].

Gedimų medžiai [13,14] yra grafinė metodika, kuri schematiškai parodo kokie sistemos įvykiai gali sukelti nepageidaujamą efektą. Šis metodas gali apjungti techninius gedimus ir žmogiškąjį faktorių.



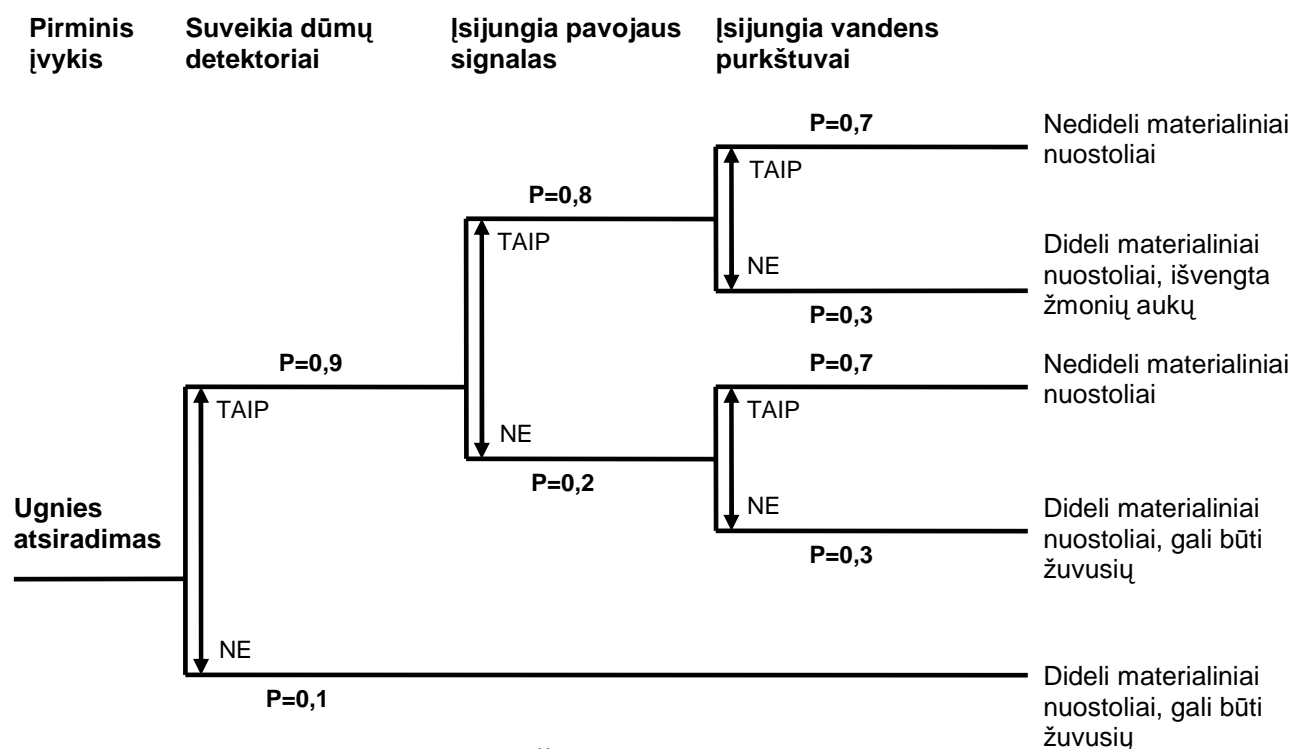
1.2 pav. Automobilių susidūrimo sankryžoje gedimų medis

Kiekvienas sistemos gedimas ar žmogaus poelgis vaizduojamas kaip tam tikra medžio viršūnė. Pats pavojingiausias ir labiausiai nepageidaujamas įvykis, toks kaip sproginimas ar užterštumo pasklidimas, vadinamas svarbiausiu įvykiu (angl. top event). Jis vaizduojamas aukščiausia medžio viršūne. Tolimesnis gedimų medis yra sudaromas išskaidant aukščiausią įvykį į įvykių sekas ar jų kombinacijas, kurios gali sukelti pavojingiausią situaciją. Kiekviename medžio konstravimo žingsnyje loginiais operatoriais aukštesnis įvykis yra išskaidomas į paprastesnius įvykius tol kol yra pasiekiamos pirminės įvykio priežastys. Jeigu yra žinomos pradinių įvykių tikimybės, galima įvertinti aukščiausio įvykio atsiradimo tikimybę.

Kaip pavyzdį nagrinėkime automobilių susidūrimo sankryžoje tikimybės paskaičiavimo problemą. Šio uždavinio gedimų medis pavaizduotas 1.2 paveiksle. Kaip matome, susidūrimo tikimybė yra 0,001.

Įvykių medžių analizė [13,14] yra pagrįsta binarine logika, kurioje įvykis atsiranda arba ne, elementas funkcionuoja arba neveikia. Šis metodas taikomas analizuojant pasekmes, kuriuos gali atsirasti įvykius gedimui arba kitam nepageidaujamam reiškiniui.

Įvykių medis prasideda nuo pirminio įvykio tokio kaip elemento gedimas, temperatūros, slėgio padidėjimas ar pavojingų dalelių išsiskirimas. Prieš sudarant tolimesnį įvykių medį yra nustatoma, kokios saugos sistemos gali įsijungti bei kokie papildomi gedimai gali atsirasti. Prie kiekvienos saugos sistemos ar papildomo gedimo medis išsišakoja į dvi dalis. Kiekviena šaka reiškia TAIP arba NE ir turi priskirtą tam tikrą tikimybę. Pirminio įvykio pasekmės yra nustatomos einant medžio šakomis iki galutinio rezultato.



1.3 pav. Priešgaisrinės saugos įvykių medis

1.3 paveiksle pateiktas priešgaisrinės saugos įvykių medis. Atsiradus gaisro židiniui priešgaisrinės saugos sistema su skirtingomis tikimybėmis nustato arba nenustato, kad yra gaisro pavojus. Į medį įtraukiamos vandens purškimo ir signalinės sistemos su nustatytomis nesuveikimo tikimybėmis.

Šiuo metu įvykių/gedimų medžių analizė yra viena populiariausių rizikos įvertinimo metodų – klasikinės tikimybinės saugos analizės pagrindas. Tikimybinė saugos analizė taikoma daugeliui pasaulio atominių reaktorių.

1.1.3. NAUJŲ ANALIZĖS METODŲ PLĖTOJIMAS

Tikimybinės saugos analizės metodika buvo sėkmingai taikoma skirtinguose projektuose, bet buvo pastebėta, kad taikant tikrai tokias metodikas kaip įvykių/gedimų medžių analizė, charakterizuoti tam tikras sudėtingas dinamines sistemas yra sudėtinga. Taip pat iškilė nemažai abejonių dėl įvykių/gedimų medžių analizės pilnumo ir adekvatumo sudėtingoms techninėms sistemoms. Nustatyti pagrindiniai įvykių/gedimų medžių trūkumai:

1. Binarinė elementų būsenų aibė. Analizuojamos tik dvi būsenos: sugedęs arba dirba, tačiau realiose sistemose dažnai pasitaiko ir gedimų, kuriuos reikėtų priskirti tarpinėms būsenoms. Akivaizdu, kad tokios būsenos, atitinkančios nepilnai veikiančią sistemą gali turėti svarbų indėlį kai kurių avarijų scenarijų atveju.

2. Nepriklausomumas. Dažniausiai daroma prielaida, kad gedimai yra nepriklausomi (išskyrus specialiai pagal bendrą priežastį modeliuojamus gedimus), tačiau realiose sistemos egzistuoja daug priklausomybės ryšių, kuriuos sunku įvertinti įvykių/gedimų medžiuose.

3. Senėjimo efektas. Nors šis efektas yra žinomas, tačiau jis nėra vertinamas dėl kur kas sudėtingesnio skaičiavimo algoritmo. Senėjimo įtaką galima analizuoti jautrumo analizės pagalba arba priimant konservatyvias prielaidas.

4. Laiko neįvertinimas. Įvykių/gedimų medžiais galima įvertinti įvykius tik chronologine tvarka, ją iš anksto nurodant. Realaus avarinio scenarijaus metu galimi įvairūs nukrypimai ir chronologiškai tikslus įvykių išsidėstymas yra tik prielaida. Šis trūkumas yra vienas iš svarbiausių, modeliuojant avarinius scenarijus.

5. Skaičiavimų neapibrėžtumas. Neapibrėžtumai yra neišvengiama įvykių/gedimų medžių skaičiavimo dalis, todėl tiesioginė kiekybinių rezultatų interpretacija gali būti neteisinga. Taigi kokybiniai įvykių/gedimų medžių rezultatai (avarinių scenarijų ir sekų nustatymas, silpniausių sistemos vietų radimas) yra net svarbesni, negu kiekybiniai.

Įvykių medžių analizė gali atskleisti teisingą dinaminių sistemų gedimo logiką, bet dėl proceso kintamųjų vaidmens nepaisymo, įvykių/gedimų medžiais sudėtinga modeliuoti sistemas, kuriuose atsitiktiniai įvykiai sąveikauja su nuo laiko priklausančiais fizikiniais procesais (slėgio, temperatūros kitimas). Vienintelis būdas įvykių medyje įvertinti proceso kintamąjį, yra visų susijusių proceso kintamųjų vertinimas. Kai reikia smulkiai aprašyti proceso kintamuosius arba kai kintamųjų skaičius didėja, įvykių medis gali augti labai stipriai. Be fizinio modelio įvykių medžio analizė turi apimti sąveikos tarp kintamųjų vertinimą. Todėl nustatyta tikimybė, kad bus pasiektas nepageidaujamas būvis, gali būti netiksli, o įvertinti nepageidaujamo būvio skirstinį laike yra neįmanoma.

Tikimybiniai procesai, sukelti atsitiktinių techninės/programinės įrangos gedimų ir besisiejantys su sistemos dinamika ir/arba žmogaus įsikišimu, gali sukelti reikšmingus gedimus sistemoje. Nustačius šiuos įvykių/gedimų medžių sunkumus, buvo sukurta nauja metodologija, skirta vertinti dinaminį

patikimumą ir atlikti tikimybinės dinamikos saugos analizę. Toliau pateikiamos pagrindinės charakteristikos, kada verta taikyti dinaminio patikimumo modeliavimą (Aldemir & Zio, 1998):

1. Dinaminis reiškinys turi didelį poveikį sistemos reakcijai.
2. Proceso dinamika priklauso nuo techninių sistemos gedimų ir nuo žmogaus-operatoriaus veiksmų.
3. Sistemos kitimą įtakoja sąveika tarp žmogaus-operatoriaus veiksmų ir techninių įrangos elgesio.
4. Yra daugybė situacijų, susijusių su sudėtiniais gedimo būdais ir proceso dinamika.

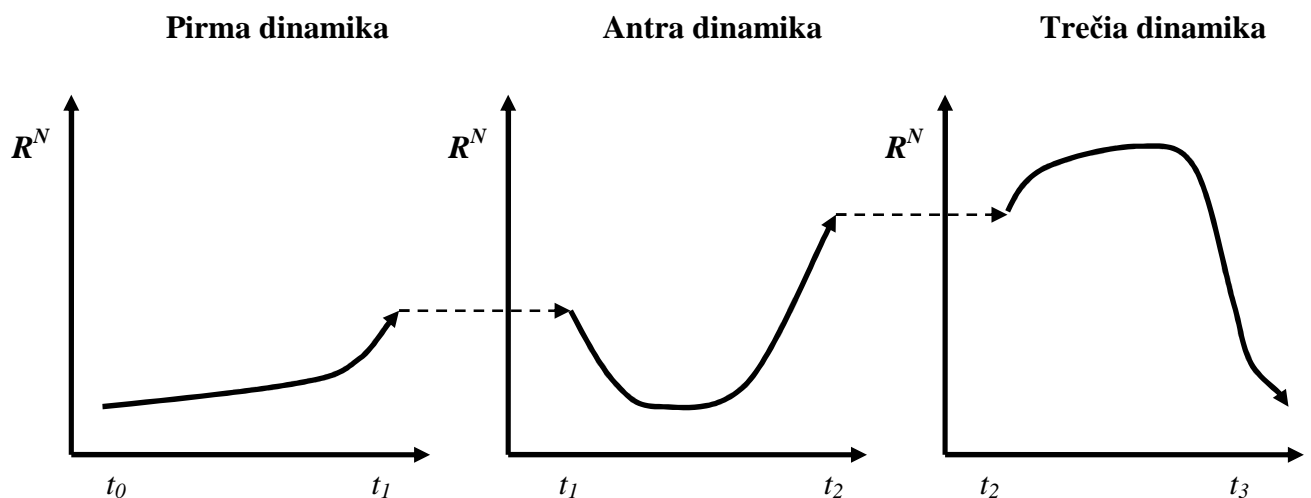
Dinaminės tikimybinės saugos analizės metodikos nėra tradicinės tikimybinės saugos analizės metodų pakaitalas. Dinaminiai metodai greičiau yra papildomi įrankiai. Remiantis dinaminės tikimybinės saugos analizės įrankiais, tyrinėtojai aiškiai supranta klasikinių požiūrių ribas ir nustato, kada dinaminiai metodai yra reikalingi [15].

1.1.4. TIKIMYBINĖS DINAMIKOS TEORIJA

Tikimybinės dinamikos teorija pateikia būdus ir metodus reikalingus analizuoti sistemas, kuriose yra sąveika tarp skirtingų fizikinių procesų (dinamikų) bei galimi atsitiktiniai perėjimai tarp jų [19]. Tarkime, kad $\bar{x} \in R^N$ yra sistemos parametrų reikšmių vektorius, o pati sistema gali būti vienoje iš d dinamikų. Sistemai esant dinamikoje i , parametrų kitimą apibūdina lygtys:

$$\frac{d\bar{x}}{dt} = \bar{q}_i(\bar{x}), \quad i = 1, 2, \dots, d. \quad (1.3)$$

Jeigu laikysime, kad i žymi esamą dinamiką, tai pora (\bar{x}, i) pilnai nusako sistemos būseną esamu momentu, o jos elgesys yra dalinai deterministinis.



1.4 pav. Dalinai deterministinis sistemos kitimas

Sistemai esant dinamikoje i parametrų reikšmės \bar{x} kinta pagal (1.3) diferencialines lygtis, tačiau šis deterministinis kitimas gali būti nutrauktas atsitiktiniu perėjimu į kitą dinamiką j . Naujoje dinamikoje sistema kinta pagal kitas diferencialines lygtis. Schematinis šio proceso vaizdas pateiktas 1.4 paveiksle.

Bendru atveju, sistemai praleidus t laiko dinamikoje i parametrų reikšmės \bar{x} galima užrašyti tokia forma:

$$\bar{x}(t) = \bar{g}_i(t, \bar{x}_0), \quad \bar{x}_0 = \bar{g}_i(0, \bar{x}_0). \quad (1.4)$$

čia \bar{x}_0 - parametrų reikšmės buvusios sistemos perėjimo į dinamiką i metu, t - laiko trukmė praleista dinamikoje i .

Perėjimai yra siejami su reikšmingu dinamikos būsenos pasikeitimu, tai yra, vienos dinamikos pabaiga ir kitos dinamikos pradžia. Kad galėtume aprašyti atsitiktinius peršokimus iš dinamikos i į dinamiką j , apibrėžiame perėjimo intensyvumus $f(i \rightarrow j / \bar{x})$, kurie apibūdina dinamikos pasikeitimą $i \rightarrow j$ bendroje įvykių sekoje. Bendras perėjimo intensyvumas paliekant dinamiką i yra lygus sumai visų perėjimų intensyvumų iš dinamikos i į bet kurią kitą dinamiką j :

$$\lambda_i(\bar{x}) = \sum_{j \neq i} f(i \rightarrow j / \bar{x}). \quad (1.5)$$

Visa informacija apie dinamiką i yra saugoma tankio funkcijoje $\pi(\bar{x}, i, t)$, kuri reiškia tikėtinumą, kad proceso kintamieji dinamikoje i laiko momentu t bus lygūs \bar{x} . Kuomet sistemoje vykstančius procesus galima sumodeliuoti kaip Markovo procesus, tuomet $\pi(\bar{x}, i, t)$ galima išreikšti Čepmeno-Kolmogorovo lygtimis:

$$\frac{\partial \pi(\bar{x}, i, t)}{\partial t} + \text{div}(\bar{q}_i(\bar{x})\pi(\bar{x}, i, t)) + \lambda_i(\bar{x})\pi(\bar{x}, i, t) = \sum_{j \neq i} f(j \rightarrow i / \bar{x})\pi(\bar{x}, i, t). \quad (1.6)$$

Integralinė $\pi(\bar{x}, i, t)$ išraiška gauta išsprendus (1.6) lygtį įgyja tokią formą:

$$\pi(\bar{x}, i, t) = \int \pi(\bar{u}, i, 0) \delta(\bar{x} - \bar{g}_i(t, \bar{u})) e^{-\int_0^t \lambda_i(\bar{g}_i(s, \bar{u})) ds} d\bar{u} + \sum_{j \neq i} \int d\bar{u} \int_0^t d\tau \pi(\bar{x}, j, \tau) f(j \rightarrow i / \bar{u}) \delta(\bar{x} - \bar{g}_i(t - \tau, \bar{u})) e^{-\int_0^t \lambda_i(\bar{g}_i(s - \tau, \bar{u})) ds}. \quad (1.7)$$

Tikimybinis tankis $\pi(\bar{x}, i, t)$ yra suma dviejų dedamųjų:

1. pirma (1.7) išraiškos dalis atitinka sistemos kitimo eigą be perėjimų laiko intervale $[0; t]$, tai yra, sistema pradiniu laiko momentu buvo dinamikoje i su parametrų reikšmėmis \bar{u} , niekada nekeitė dinamikų ir parametrų reikšmės kito pagal deterministinę dinamiką i nusakantį procesą nuo \bar{u} iki \bar{x} ;
2. antra (1.7) išraiškos dalis atitinka visas galimas situacijas, kuomet paskutinis perėjimas įvyksta iki laiko momento t ir toliau sistema kinta pagal deterministinę procesą nusakantį dinamiką i .

Perėjimams, susijusiems su dinamika i , įvedamos dvi papildomos tankių funkcijos:

- išėjimo tankis iš i dinamikos į bet kurią kitą $k \neq i$ su (\bar{x}, t)

$$\psi(\bar{x}, i, t) = \lambda_i(\bar{x})\pi(\bar{x}, i, t). \quad (1.8)$$

- įėjimo tankis į dinamiką i iš bet kurios kitos $j \neq i$ su (\bar{x}, t)

$$\varphi(\bar{x}, i, t) = \sum_{j \neq i} f(j \rightarrow i/\bar{x})\pi(\bar{x}, j, t) = \sum_{j \neq i} p(j \rightarrow i/\bar{x})\psi(\bar{x}, j, t). \quad (1.9)$$

čia $p(j \rightarrow i/\bar{x}) = \frac{f(j \rightarrow i/\bar{x})}{\lambda_i(\bar{x})}$ tikimybė, kad iš i dinamikos bus pereita į j dinamiką.

Kaip matome, abi šios funkcijos yra susijusios viena su kita per tankio funkciją $\pi(\bar{x}, i, t)$. Kai sistemoje vykstantys procesai gali būti apibūdinami kaip Markovo procesai, $\pi(\bar{x}, i, t)$ gali būti išreikšta tiek per įėjimo tankį $\varphi(\bar{x}, i, t)$, tiek per išėjimo tankį $\psi(\bar{x}, i, t)$ [19]:

$$\begin{aligned} \pi(\bar{x}, i, t) &= \int \pi(\bar{u}, i, 0) \delta(\bar{x} - \bar{g}_i(t, \bar{u})) e^{-\int_0^t \lambda_i(\bar{g}_i(s, \bar{u})) ds} d\bar{u} + \\ & \sum_{i \neq j} \int_0^t d\tau \int d\bar{u} \psi(\bar{u}, j, \tau) p(j \rightarrow i/\bar{u}) \delta(\bar{x} - \bar{g}_i(t - \tau, \bar{u})) e^{-\int_0^t \lambda_i(\bar{g}_i(s, \bar{u})) ds} \\ \pi(\bar{x}, i, t) &= \int \pi(\bar{u}, i, 0) \delta(\bar{x} - \bar{g}_i(t, \bar{u})) e^{-\int_0^t \lambda_i(\bar{g}_i(s, \bar{u})) ds} d\bar{u} + \\ & \int_0^t d\tau \int d\bar{u} \varphi(\bar{u}, j, \tau) \delta(\bar{x} - \bar{g}_i(t - \tau, \bar{u})) e^{-\int_0^t \lambda_i(\bar{g}_i(s, \bar{u})) ds} \end{aligned} \quad (1.10)$$

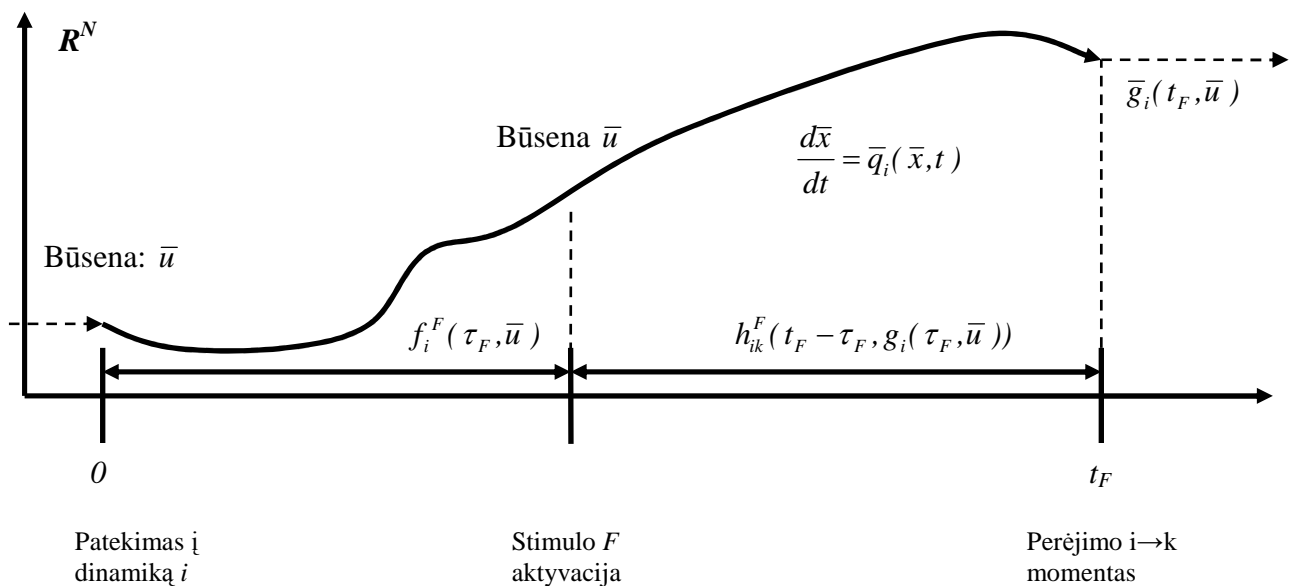
1.1.5. STIMULIUOJAMOS DINAMIKOS TEORIJA

Tikimybinės dinamikos teorija pateikia būdus ir metodus reikalingus analizuoti sistemas, kuriose yra sąveika tarp skirtingų fizikinių procesų (sistemos dinamikų) bei galimi atsitiktiniai perėjimai tarp jų. Naudojant tikimybinės dinamikos teoriją, remiamasi prielaida, kad perėjimai tarp dinamikų vyksta iš karto, kai tik sistemos parametrai pasiekia tam tikrą ribą. Tačiau realiose sistemose perėjimo procesas priklauso nuo dviejų perėjimą įtakojančių atsitikimų (veiksnių). Pirmasis atsitikimas yra tam tikros sąlygos, kuri gali sukelti dinamikos pasikeitimą, tenkinimas. Šį atsitikimą vadinsime stimulo aktyvacija. Po to, kai aktyvuojamas stimulus, gali susidaryti tam tikras laiko uždelsimas kol įvyks pats įvykis, sukeliantis perėjimą iš vienos dinamikos į kitą. Uždelsimo pabaiga laikoma antru perėjimą įtakojančiu atsitikimu.

Stimulo aktyvaciją ir uždelsimą pailiustruosime paprastu buitiniu pavyzdžiu. Tarkime, kad virtuvėje netyčia ir nepastebėtai atsukama dujinės viryklės sklendė, dėl to pradeda tekėti gamtinės dujos. Po kažkiek laiko virtuvėje esantis dujų mišinys taps degus ir atsitiktinai atsiradusi kibirkštis sukels sprogią. Analizuojant šią situaciją, stimulo aktyvacija laikomas sprogios dujų koncentracijos pasiekimas, laiko trukmė tarp sprogios dujų koncentracijos ir kibirkšties atsiradimo vadinama uždelsimu. Pats sprogimas laikomas reikšmingu dinamikos pasikeitimu, kuris priklauso nuo stimulo aktyvacijos ir uždelsimo trukmės.

Uždaviniuose, kuriuose laiko uždelsimo negalima nepaisyti ir uždelsimo trukmė yra atsitiktinis dydis, tikimybinės dinamikos teorija negali būti taikoma, todėl yra vystoma stimuliuojamos dinamikos teorija, kuri yra tikimybinės dinamikos papildymas [19].

Prieš nagrinėjant pagrindinius stimuliuojamos dinamikos elementus, apibrėšime pagrindines sąvokas susijusias su įvykiais. Įvykis yra siejamas su momentiniu dinamikos pasikeitimu, kurį sukėlė stimulo aktyvacija, atskirtu atveju, įvykis e reiškia dinamikos i pasikeitimą į dinamiką j laiko momentu t . Atsitiktiniu įvykiu laikomas įvykis, kurio atsiradimas dėl sudėtingos prigimties modeliuojamas stochastiškai. Determinuoto įvykio atsiradimas nusakomas apibrėžtomis taisyklėmis ar sąlygomis. Tokio įvykio pavyzdys gali būti automatinis apsaugos sistema įsijungimas, kai sistemos slėgio ar temperatūros reikšmės pasiekia tam tikrą ribą. Dinamikų pasikeitimai arba proceso parametru kitimas tarp perėjimų yra siejami su determinuotais pereinamaisiais procesais. Dinamikų pasikeitimai priklauso nuo pradinių sąlygų, susijusių su procesų parametru reikšmėmis pradiniu laiko momentu. Perėjimu laikomas reikšmingas dinamikos pasikeitimas, t.y. vienos dinamikos pabaiga ir kitos dinamikos pradžia. Perėjimų sekos yra susijusios su įvykių sekomis, dalis jų gali būti dirbtinai sukuriamos imitacinio modeliavimo tikslais.



1.5 pav. Sistemos kitimas dinamikoje i įvykius stimulo F aktyvacijai

Pirminiuose tikimybinės dinamikos teorijos uždaviniuose laikoma, kad dinamikos pasikeitimas įvyksta iš karto po įvykio, kurio pasekmės gali sukelti perėjimą į kitą sistemos būseną. Stimuliuojamos tikimybinės dinamikos teorijos pagrindinė idėja yra stimulo aktyvacijos, turinčios įvykti prieš dinamikų pasikeitimą, įvedimas. Stimulu galima apibūdinti bet kokią situaciją, kuri po tam tikro laiko potencialiai gali sukelti įvykį, kurio metu pakeičiama sistemos dinamika.

Tegul Γ yra stimulų galinčių sukelti tam tikrą įvykį aibė. Laikykime, kad $f_i^F(\tau_F, \bar{u})$ yra stimulo $F \in \Gamma$ aktyvavimo tankio funkcija, kai sistema yra $\bar{g}_i(\tau_F, \bar{u})$ būsenoje, čia τ_F žymi laiko trukmę praleistą i -oje dinamikoje, o \bar{u} yra sistemos parametrų reikšmės, kurioms esant buvo pereita į šią dinamiką. Taip pat apibrėžkime $h_{ik}^F(t_d, \bar{a})$ - tikimybę per laiko vieneta, kad tarp stimulo aktyvacijos ir įvykio atsiradimo, tai yra perėjimo į dinamiką k , bus laiko uždelsimas $t_d = t_F - \tau_F$, kai stimulus buvo aktyvuotas esant būsenai $\bar{a} = \bar{g}_i(\tau_F, \bar{u})$, o $h_i^F(t_d, \bar{u}) = \sum_{k \neq i} h_{ik}^F(t_d, \bar{u})$ yra uždelsimo t_d tankio funkcija, kur t_d yra laiko skirtumas tarp laiko momentų, kai stimulus F aktyvuojamas ir išeinama iš dinamikos i .

Aprašytos stimulo aktyvacijos ir uždelsimų tankio funkcijos yra sudarytos laikantis prielaidos, kad sistemai perėjus į naują dinamiką visi ankščiau aktyvuoti stimulai yra deaktyvuojami, tai yra, ankstesni stimulai negali sukelti naujų įvykių, o sistemos tolimesnis vystimasis naujoje dinamikoje vyksta užmirštant viską, kad buvo prieš pereinant į ją. Ši prielaida ne visada yra priimtina. Tam tikri įvykiai atsitinka praėjus kažkiek nuo stimulo aktyvacijos nepriklausomai nuo to, kurioje dinamikoje tuo momentu yra sistema. Taigi tam tikra su laiku susisiejusi stimulų informacija turėtų būti visada išsaugota.

Kad įgyvendintume šį ne Markovo modelį, sudaroma sąlyginė uždelsimų tankio funkcija \tilde{h}_i^F , kuri taikoma, kai stimulus F buvo aktyvuotas sistemai esant ankstesnėje dinamikoje j ir išliko aktyvus po perėjimo į dinamiką i [19].

Tegul τ yra laiko momentas, kada sistema perėjo į dinamiką j su parametrų reikšmėmis \bar{u} , $\tau_F < \tau$ - laiko momentas, kada buvo aktyvuotas stimulus F . Nagrinėsime skirtingas situacijas su skirtingais uždelsimais. Pačiu paprasčiausiu atveju, kada uždelsimo tankio funkcija priklauso tik nuo laiko, uždelsimo susijusio su stimulu F tankio funkcija h^F :

$$h^F(t - \tau_F; j \rightarrow i, \tau) = \begin{cases} h_j^F(t - \tau_F) & \tau_F \leq t \leq \tau \\ (1 - H_j^F(\tau - \tau_F)) \frac{h_i^F(t - \tau_F)}{1 - H_i^F(\tau - \tau_F)} & t \geq \tau \end{cases} \quad (1.11)$$

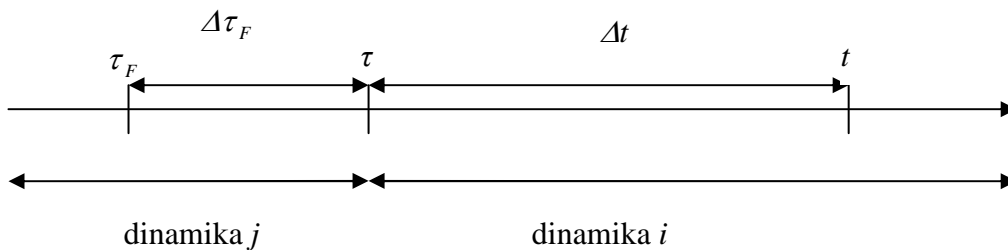
Čia $H_j^F(\tau - \tau_F)$ yra stimulo F uždelsimų pasiskirstymo funkcija dinamikoje j . Formulės (1.11) antra eilutė yra sudaryta tokiu principu: uždelsimas yra ilgesnis negu $\tau - \tau_F$, jeigu jis nesibaigė sistemai esant būsenoje j , kitaip tariant, su tikimybe $1 - H_j^F(\tau - \tau_F)$ įvykis neatsitiko dinamikoje j , o perėjus į dinamiką i , naudojama uždelsimų tankio funkcija h_i^F su sąlyga, kad uždelsimas yra didesnis negu $\tau - \tau_F$. Toks tankio funkcijos sudarymas yra galimas, jeigu laikoma, kad perėjimas iš j

dinamikos į dinamiką i nepakeičia uždelsimo trukmės, kitaip tariant, uždelsimo pabaiga galima dinamikose i ir j .

Nagrinęjant sąlyginę išliekančių uždelsimų tankio funkciją $\tilde{h}_i^F(\Delta t / \Delta \tau_F)$, kuri reikštų įvykio atstiktumą dinamikoje i praėjus $\Delta t = t - \tau$ laiko nuo perėjimo į šią dinamiką, kai nuo stimulo aktyvacijos iki perėjimo į dinamiką i praėjo $\Delta \tau_F = \tau - \tau_F$ laiko, gauname tokią išraišką:

$$\tilde{h}_i^F(\Delta t / \Delta \tau_F) = \frac{h^F(t - \tau_F; j \rightarrow i, \tau)}{1 - H_j^F(\Delta \tau_F)} = \frac{h_i^F(\Delta t + \Delta \tau_F)}{1 - H_i^F(\max(0, \Delta \tau_F))}. \quad (1.12)$$

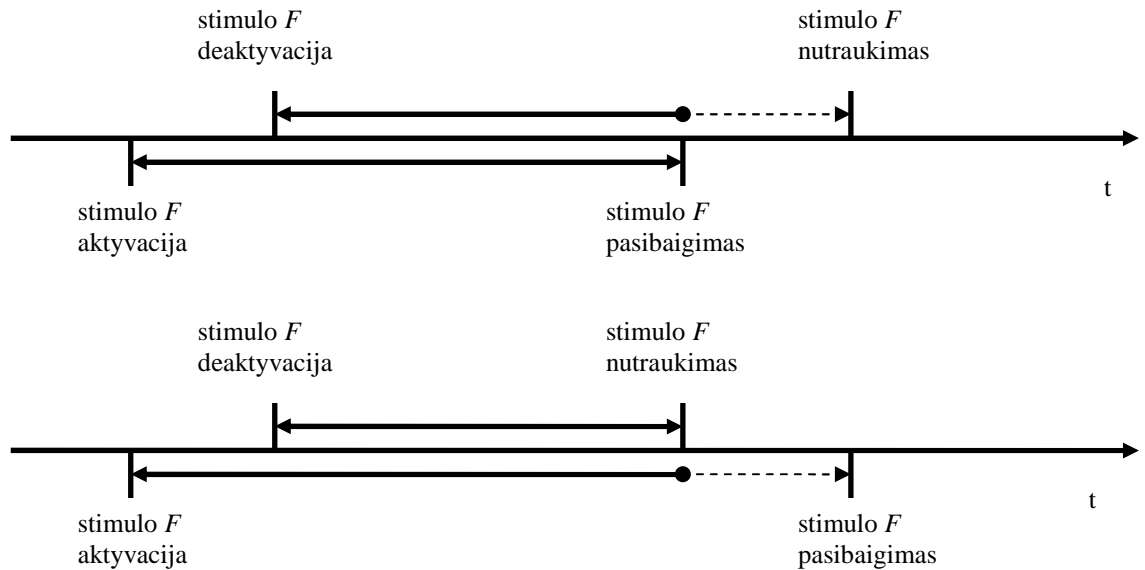
Bendru atveju, nuo stimulo aktyvacijos iki įvykio, kurį sukelia tas stimulus praeina $\Delta \tau_F + \Delta t$ laiko, bet jeigu aktyvacija išlieka perėjus į naują dinamiką i , uždelsimo skirstinys i dinamikoje turi būti nupjautas iki $\Delta \tau_F$ momento [19].



1.6 pav. Išliekantys uždelsimai po perėjimo į naują dinamiką

Sudarant stimuliuojamų dinamikų modelį turime atsižvelgti į skirtingas situacijas, kurias gali sukelti stimulų aktyvacijos. Tegul Γ - yra visi galimi stimulai, A - visi aktyvuoti stimulai, tuomet yra galimos tokios su stimulu susijusios situacijos:

- Stimulo aktyvacija. Naujo stimulo $F \in \Gamma / A$ aktyvacija. Stimulus F įtraukiamas į aibę A .
- Stimulo uždelsimo pasibaigimas. Uždelsimo, susijusio su stimulu $F \in A$, pabaiga. Atsiranda įvykis, kurį sukėlė stimulus $F \in A$. F pašalinamas iš aibės A .
- Stimulo deaktyvacija. Stimulo $F \in A$ deaktyvacijos iniciacija. F įtraukiamas į aibę D . D – aibė stimulų, kuriems yra pradėta deaktyvacija.
- Stimulo nutraukimas. Uždelsimo susijusio su stimulo $F \in D$ deaktyvacija pabaiga. Stimulus F šalinamas iš aibių D ir A .



1.7 pav. Galimos su stimulais susijusios situacijos

Sistemos perėjimas į kitą dinamiką įvyks tik tai tuomet, kai stimulo uždelsimo pabaiga atsiras anksčiau negu stimulo nutraukimas. Visais kitais atvejais perėjimas neįvyks.

Analizuojant skirtingas būsenas turime sudaryti skirtingas įėjimo tankio funkcijas, kurios sieja su kiekviena galinčia įvykti būseną. Pirmiausia analizuosime būsenas, susijusias su stimulo aktyvavimu

- **Stimulo aktyvacija.** $\varphi_F(\bar{x}, j, t, \tau, \bar{\tau}_A, A)$ - stimulo F aktyvavimo tankio funkcija dinamikoje j , esant parametru reikšmėms \bar{x} laiko momentu t . τ žymi laiko momentą, kada buvo pereita į j dinamiką, A – aktyvuotų stimulų aibė, o $\bar{\tau}_A$ saugo stimulų aktyvacijų laiko momentus. Atsitikus stimulo F aktyvacijai laiko momentu t , F įtraukiamas į aibę A , o t į vektorių $\bar{\tau}_A$, kur $t > \tau_G, \forall G \in A, G \neq F$. Aktyvacijos tankio funkcija turi būti naudojama sistemingai kartu su Dirako piko funkcija $\delta(t - \tau_F)$, kada parametrai t ir τ_F yra nepriklausomi.

$$\varphi_F(\bar{x}, j, t, \tau, \bar{\tau}_A, A + \{F\}) = \int d\bar{u} \int_{\tau}^t d\tau^* \delta(\bar{x} - \bar{g}_j(t - \tau^*, \bar{u})) p_j^{F^*}(t, \tau^*, \tau, \bar{u}, \bar{\tau}_A, A) \times \left[\pi(\bar{u}, j, \tau) \delta(\tau) \delta_{A,0} + \varphi_{in}(\bar{u}, j, \tau^*, \bar{\tau}_A, A) \delta(\tau^* - \tau) + \sum_{G \in A} \varphi_G(\bar{u}, j, \tau^*, \tau, \bar{\tau}_A, A) \delta(\tau^* - \tau_G) \right] \quad (1.13)$$

čia $p_j^{F^*}(t, \tau^*, \tau, \bar{u}, \bar{\tau}_A, A)$ yra tikimybė, kad stimulus F bus kitas aktyvuotas stimulus dinamikoje j laiko intervale $[t, t + dt]$, kai stimulai, priklausantys aibei A , išlieka aktyvūs sistemai perėjus į dinamiką i

$$p_i^{F^*}(t, \tau^*, \tau, \bar{u}, \bar{\tau}_A, A) dt = \frac{f_i^F(t - \tau) dt}{1 - F_i^F(\tau^* - \tau, \bar{u})} \cdot \prod_{\substack{H \in A \\ H \neq F}} \frac{1 - F_i^H(t - \tau, \bar{u})}{1 - F_i^H(\tau^* - \tau, \bar{u})} \times \prod_{G \in A} \frac{1 - \tilde{H}_i^G(t - \tau, \bar{u} / \Delta\tau_G)}{1 - \tilde{H}_i^G(\tau^* - \tau, \bar{u} / \Delta\tau_G)} \quad (1.14)$$

• **Įvykio atsiradimas.** $\varphi_{in}(\bar{x}, j, t, \bar{\tau}_A, A)$ - įėjimo į dinamišką j tankio funkcija, kai sistemos parametrų reikšmės yra \bar{x} laiko momentu t . Laikoma, kad pereinant į naują dinamišką j visi aktyvuoti stimulai, priklausantys aibei A , išlaiko savo aktyvią būseną. Jeigu nenagrinėsime papildomų stimulų aktyvacijų, kurias gali sukelti perėjimas į kitą dinamiką, tuomet t yra viršutinė riba visiems vektorių $\bar{\tau}_A$ elementams.

$$\begin{aligned} \varphi_{in}(\bar{x}, j, t, \bar{\tau}_A, A) &= \\ &= \sum_{A' \supset AF \in A' i \neq j} \sum_{\bar{u}} \int_0^t d\bar{u} \int_0^t d\tau^* \int_0^{\tau^*} d\tau \int_0^{\tau^*} \dots \int_0^{\tau^*} d\bar{\tau}_{A'/A} \delta(\bar{x} - \bar{g}_i(t - \tau^*, \bar{u})) p_{ij}^F(t; \tau^*, \tau, \bar{u}, \bar{\tau}_{A'}, A') \\ &\times \left[\varphi_{in}(\bar{u}, i, \tau^*, \bar{\tau}_{A'}, A') \delta(\tau^* - \tau) + \sum_{G \in A'} \varphi_G(\bar{u}, i, \tau^*, \bar{\tau}_{A'}, A') \delta(\tau^* - \tau_G) \right] \delta_{ij}^F(A' \rightarrow A) \end{aligned} \quad (1.15)$$

čia $p_{ij}^F(t; \tau^*, \tau, \bar{u}, \bar{\tau}_{A'}, A')$ - tikimybė, kad su aktyvuotu stimulu F susijęs uždelsimas pasibaigs laiko intervale $[t, t + dt]$ ir sistema pereis į naują dinamišką j prieš atsitinkant bet kuriai kitai stimulų būsenai. Laikoma, kad aktyvių stimulų aibė A išlieka tokia pat po dinamikų pasikeitimo.

$$\begin{aligned} p_{ij}^F(t, \tau^*, \tau, \bar{u}, \bar{\tau}_A, A) dt &= \\ &= \frac{\tilde{h}_{ij}^F(t - \tau / \Delta\tau_F) dt}{1 - \tilde{H}_i^F(\tau^* - \tau, \bar{u} / \Delta\tau_F)} \cdot \prod_{\substack{G \in A \\ G \neq F}} \frac{1 - \tilde{H}_i^G(t - \tau, \bar{u} / \Delta\tau_G)}{1 - \tilde{H}_i^G(\tau^* - \tau, \bar{u} / \Delta\tau_G)} \times \prod_{F \in A} \frac{1 - F_i^H(t - \tau, \bar{u})}{1 - F_i^H(\tau^* - \tau, \bar{u})} \end{aligned} \quad (1.16)$$

Nagrinėjant būsenas, susijusias su stimulų deaktyvacija, atitinkamai deaktyvacijos būsenai ir nutraukimas gali būti pateikti per pseudo atvirkštinius stimulus. Tarkim F^- yra fiktyvus stimulus, susietas su stimulu F , laikykime, kad stimulo F^- aktyvacijos tankio funkcija yra tapati F stimulo deaktyvacijos tankio funkcijai. Taip pat laikome, kad su stimulu F^- susijęs fiktyvus įvykis turi nulinių uždelsimą. Jeigu sistema yra dinamikoje i , tai fiktyvus įvykis atlieka perėjimą į tą pačią sistemą i . Kitaip tariant, laikome, kad deaktyvacija vyksta akimirksniu be uždelsimų bei vykdo perėjimą iš dinamikos i į i . Tokiu atveju, F ir F^- bus deaktyvuoti, o kiti stimulai priklausys aibei A .

1.2. NEAPIBRĖŽTUMO ANALIZĖS METODOLOGIJA

1.2.1. NEAPIBRĖŽTUMO ANALIZĖS TIKSLAI IR TAIKYMAS

Kiekvienas sudėtingos techninės sistemos modelis neišvengiamai turi tam tikrą neapibrėžtumą. Nepastovumas ir neapibrėžtumas gali būti sąlygoti įvairių priežasčių ir šaltinių, kurie dar skirstomi į tokias tris grupes:

1. Pradinių duomenų ir parametrų reikšmių išsibarstymas. Daugelio parametrų matavimų rezultatai yra stochastiniai ir nepriklauso nuo to, kaip dažnai jie yra matuojami, siekiant padidinti žinių lygį apie juos. Pradinių duomenų neapibrėžtumai taip pat susiję su duomenų trūkumu, daromomis modeliavimo prielaidomis ar didele duomenų sklaida. Tikimybinuose modeliuose dėl duomenų trūkumo naudojami apibendrinti ir specifiniai sistemos duomenys, kurie į ieškomą rezultatą gali

įtraukti nepastovumą ir neapibrėžtumą. Duomenų neapibrėžtumas pasižymi tuo, kad atliekant modelio duomenų neapibrėžtumų analizę jis dažnai gali būti įvertintas kiekybiškai.

2. Modelio nepilnumas. Ši neapibrėžtumų rūšis kyla dėl galimų modelio pradinių įvykių ir scenarijų pilnumo trūkumo. Nepilnumą yra sunkiausia įvertinti kiekybiškai. Dažniausiai naudojamas pradinių įvykių sąrašo pilnumo tikrinimo metodas, kuris pagrįstas patikra, ar šis įvykių sąrašas padengia visas galimas situacijas.

3. Modeliavimo neadekvatumas. Modelio ir realaus objekto neadekvatumai gali atsirasti visose modelio sudarymo ir modeliavimo stadijose, nes kuriant konceptualinį modelį dažnai daromi tam tikri suprastinimai, būsenų apjungimai. Į modelį įtraukti scenarijai ir sekos nėra visiškai adekvatūs dėl įvairių prielaidų, suprastinimų, matematinių modelių trūkumų, korelacių netikslumo ar ribotų tikslaus skaičiavimo galimybių. Siekiant įvertinti galimą modeliavimo prielaidų įtaką, atliekama jautrumo analizė.

Neapibrėžtumai yra būdingi beveik visiems sistemų modeliavimo rezultatams. Nors neapibrėžtumų įvertinimas ir labai apsunkina sistemos modeliavimą, tačiau jis būtinas siekiant nustatyti sistemos modelio rezultatų adekvatumą ir ribotumus.

Bendru atveju, gali būti išskirtos tokios neapibrėžtumo analizės taikymo sritys:

- Nepatikimiausių modelio elementų nustatymas. Galima identifikuoti tuos modeliavimo proceso elementus, kurie yra mažiausiai patikimi ir kuriems reikalinga daugiausia dėmesio didinant jų patikimumą. Neapibrėžtumo analizė padeda identifikuoti sritis, kuriose yra duomenų ir kitos informacijos trūkumas, taip pat gali padėti įvertinant ir gerinant programinės įrangos, bei jos pagalba sudarytų modelių kokybę.

- Sprendimų priėmimo pagrįstumas. Atliktų skaičiavimų rezultatų pagrindu priimant svarbius sprendimus (pvz. avarių valdymo planavimas), paprastai atsižvelgiama į tuos duomenų neapibrėžtumus, kurie gali sąlygoti alternatyvių sprendimų priėmimą. Neapibrėžtumo analizė šiuo atveju įgalina priimti labiau pagrįstus sprendimus ir nustatyti tas nagrinėjamos problemos sritis, kuriose yra reikalinga turėti daugiau žinių.

- Geriausio įverčio panaudojimas. Rezultatų neapibrėžtumo analizė yra taikoma vykdant pavojų ir rizikos analizę, nustatant saugos kriterijus, atliekant pavojingų objektų licencijavimą. Adekvatesni ir efektyvesni saugos lygio vertinimo ir ribų nustatymo modeliai yra tie, kurie pagrįsti ne tiek konservatyviomis prielaidomis, kiek geriausio įverčio skaičiavimais, taikomais kartu su neapibrėžtumo analizės rezultatais.

Šiuo metu pasaulyje naudojama keletas įvairių neapibrėžtumo vertinimo teorijų. Klasikine yra laikoma Bajeso tikimybių teorija [2] bei galimybių teorija [9]. Pati bendriausia neapibrėžtumo vertinimo teorija yra pagrįsta Sudženo (angl. Sugeno) [23] neraiškumo matais. Kitos teorijos yra

gautos iš Sudženo neraiškumo įverčio teorijos pakeitus neraiškumo matą labiau apribotomis aksiomomis.

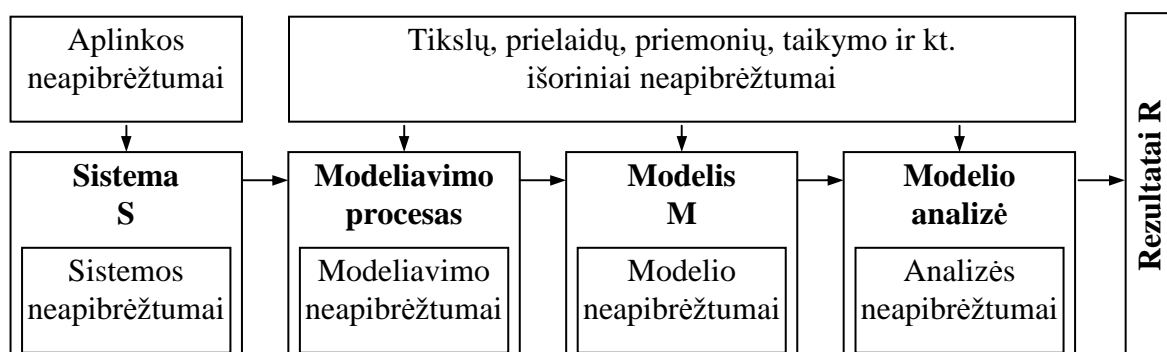
Keletas žinomų teorijų:

- akivaizdumo (angl. k. evidence);
- galimybių [9];
- robastinė patikimumo [3];
- intervalų aritmetika.

Nežiūrint teorijų įvairovės, neapibrėžtumai dažniausia apibūdinami naudojant tikimybių teoriją. Tikimybiniai metodai yra susiję su atsitiktiniais kintamaisiais, kurie naudojami aprašant neapibrėžtumo šaltinius ir, bendru atveju, yra siejami su patikimumo metodais, kurie naudojami, kai nagrinėjama pakankamai mažo sudėtingumo (iki 100 atsitiktinių kintamųjų) ir pakankamai gerai suprantama sistema. Neapibrėžtumų modeliavimui naudojami galimybių teorijos metodai yra naudingi, kai nagrinėjamos didelės, sudėtingos sistemos arba tokios sistemos, kurios dėl savo prigimties negali būti įvertintos statistiškai. Galimybių teorija yra taikytina ten, kur modeliavimas pagrįstas statistiniais duomenimis yra nepraktiškas arba, netgi ir neįmanomas.

1.2.2. NEAPIBRĖŽTUMŲ ĮTAKA MODELIAVIMO REZULTATAMS

Atliekant neapibrėžtumo analizę, pradiniu žingsniu laikoma neapibrėžtumų šaltinių identifikavimas ir jų apibūdinimas. Šiam tikslui neapibrėžtumai klasifikuojami pagal jų atsiradimo priežastis ir potencialias pasekmes. Neapibrėžtumų kilmė gali būti siejama su sistemos modelio parametrų matavimo paklaidomis bei informacijos trūkumu arba natūralia, bet nevaldoma parametrų variacija. Nagrinėjant sudėtingos sistemos modelio analizės rezultatų neapibrėžtumą reikia atsižvelgti į visą rezultatų atsiradimą sąlygojančią seką (1.8 paveikslas), kurią sudaro sistema, modeliavimo procesas, modelis ir modelio analizė [1].



1.8 pav. Sistemos modelio rezultatų neapibrėžtumą įtakojanti seka

Atsižvelgus į neapibrėžtumų valdymo galimybes išskiriami išoriniai ir vidiniai neapibrėžtumų šaltiniai bei atitinkami modelio parametrai. Jeigu neapibrėžtumų šaltinių (kraštinių sąlygų, parametrų

ir kt.) valdymo galimybės, labai mažos arba jų nėra, tai šie neapibrėžtumai laikomi išoriniais. Potencialios neapibrėžtumų pasekmės siejamos su rezultatų jautrumo analize bei rezultatams ir modelio parametrams taikomais priimtino kriterijais.

Sistemos ir jos aplinkos parametrų bei su modeliu susijusių vidinių parametrų neapibrėžtumo įtaka rezultatui gali būti įvertinta kiekybiškai [17]. Matematinis modelius galima aprašyti kaip funkciją:

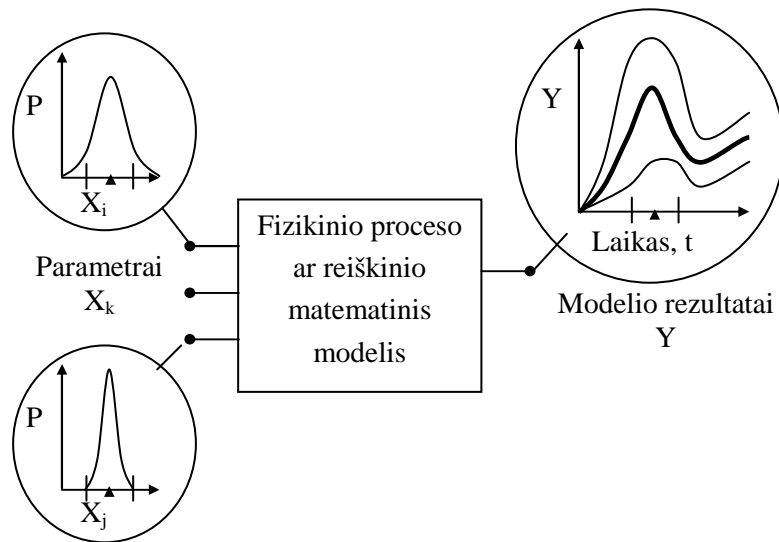
$$y = F(x_1, x_2, \dots, x_N) \quad (1.17)$$

čia x_1, x_2, \dots, x_N - modelio parametrai, N - parametrų kiekis, y - modelio rezultatas, $F(\cdot)$ - funkcija siejanti modelio parametrus ir rezultatus. Funkcija $F(\cdot)$ nebūtinai turi analitinę išraišką, sudėtingų modelių atveju, funkciją $F(\cdot)$ galime laikyti programų paketo skaičiavimus aprašančia funkcija.

Matematinio modelio aprašymas (1.17) yra suprastintas, nes realiuose modeliuose dažniausiai turime ne vieną, o daug rezultatų, apibūdinančių įvairias nagrinėjamo proceso ar reiškinio charakteristikas. Taip pat dažnai turime analizuoti rezultatus, kurie priklauso nuo laiko t , t.y. $y = y(t)$, todėl funkcija $F(\cdot)$ taip pat turėtų priklausyti nuo laiko. Tačiau suprastintas modelio aprašymas nekeičia neapibrėžtumo ir jautrumo analizės principų, kadangi jeigu turime daug modelio rezultatų (pvz., slėgis, temperatūra, srauto debitas) neapibrėžtumo analizę galima būtų atlikti kiekvienam rezultatui atskirai ar tik svarbiausiems. Dažnai modelio rezultatai skaičiuojami įvairiems laiko momentams, tokiu atveju neapibrėžtumo ir jautrumo analizę reikėtų atlikti kiekvienam laiko momentui atskirai, arba pasirinktiems charakteringiems taškams, pvz., maksimaliai reikšmei.

Modelio rezultato neapibrėžtumo problema kyla dėl to, kad modelio parametrai x_1, x_2, \dots, x_N nėra tiksliai žinomi arba gali keistis realios aplinkos sąlygomis. Todėl modelio parametrus galime laikyti atsitiktiniais dydžiais ir juos aprašyti tikimybiniais skirstiniais $p_1(x_1), p_2(x_2), \dots, p_N(x_N)$. Paprasčiausiu atveju, kai parametras yra išmatuojamas dydis ir jo reikšmės neapibrėžtumas kyla tik dėl matavimo paklaidos, paprastai naudojamas normalusis (Gauso) tikimybinis skirstinys. Praktikoje dar taikomi tolygusis, trikampus, lognormalusis, beta ir kt. skirstiniai. Pasirinktas tikimybinis skirstinys turi kiek galima geriau atspindėti esamas žinias apie galimą parametro reikšmę.

Modelio parametrų neapibrėžtumo schematinis ryšys su modelio rezultatų neapibrėžtumu pavaizduotas .



1.9 pav. Modelio parametrų neapibrėžtumo ryšys su rezultatų neapibrėžtumu

Žinodami modelio parametrų tikimybinis skirstinius galime atsitiktinai parinkti parametrų reikšmes ir sudaryti modelio parametrų reikšmių rinkinius. Atlikus modelio skaičiavimus kiekvienam parametrų rinkiniui gaunamos skirtingos modelio rezultato reikšmės ir tai atspindi rezultato neapibrėžtumą. Siekiant įvertinti modelio rezultatų neapibrėžtumą kiekybiškai, modelio rezultatas analizuojamas kaip atsitiktinis dydis ir vertinamos modelio rezultato skirstinio charakteristikos: vidurkis, standartinis nuokrypis, kvantiliai bei nustatomi rezultatų pasikliautinumo intervalai.

1.2.3. MODELIO NEAPIBRĖŽTUMO ĮVERTINIMAS

Dažniausiai taikomas modeliavimo neapibrėžtumas vertinimas naudojant tikimybinis skirstinius. Naudojant Monte Karlo tipo metodus gali būti gaunama modelio rezultatų pasiskirstymo funkcija, kuri įgalina įvertinti vidurkį, medianą kvantilius, pasikliautinumo intervalus ir kitas tikimybinės rezultato charakteristikas [12,18]. Tuo pačiu metu gali būti gaunami parametrų ir rezultatų koreliacijos bei regresijos koeficientai [6,7], kurie įgalina atlikti modelio rezultatų jautrumo analizę.

Atliekant modelio rezultatų neapibrėžtumo analizę, pirmiausia reikėtų įvertinti modelio rezultatų skaičiavimo laiką. Jei šis laikas nėra ilgas (kelios minutės ar trumpiau), galima sudaryti didelę (1000 ir daugiau) modelio parametrų reikšmių rinkinių atsitiktinę imtį bei gauti tokį patį skaičių modelio rezultato reikšmių. Esant tokio dydžio imčiai, įprastų statistinių metodų pagalba galime pakankamai tiksliai nustatyti modelio rezultato skirstinio tipą bei įvertinti jo charakteristikas.

Norint įvertinti galimą modeliavimo rezultatų neapibrėžtumą, įprastai yra apskaičiuojamas statistinės tolerancijos intervalas ar vienpusė tolerancijos riba [11]. Tolerancijos intervalą apibrėžia dvi tikimybinės charakteristikos (α , β), kur α – parametras, nusakantis intervalą tarp atsitiktinio dydžio skirstinio kvantilių, apimančio ne mažiau $100\% \cdot \alpha$ skirstinio reikšmių, o β – pasiklovimo lygmuo, savo prasme atitinkantis statistikoje gerai žinomą pasiklovimo lygmenį (tikslumo

tikimybe). (α, β) tolerancijos intervalo interpretacija yra tokia - su tikimybe β , į tolerancijos intervalą patenka bent $100\% \cdot \alpha$ visų modelio rezultato reikšmių.

Jeigu modelio skaičiavimo laikas yra ilgas ir atlikti daug skaičiavimų reikia didelių kompiuterinių resursų. Tuomet norimas modelio rezultato skirstinio charakteristikas galime vertinti naudodami mažesnę atsitiktinę imtį. Anglų statistikas S.S. Wilks 1941 metais paskelbė darbą [4], kuriame pateikė metodą, kaip iš nedidelės paprastosios atsitiktinės imties galima paskaičiuoti (α, β) tolerancijos intervalą. Wilks pateiktas metodas leidžia įvertinti, kokį minimalų modelio skaičiavimų kiekį reikia atlikti, kad suformuoti imtį, kurios maksimali ir minimali modelio rezultato reikšmė atitiktų norimo tikimybinio lygio intervalo galus. Praktikoje svarbi Wilks metodo savybė yra tai, kad reikalingų atlikti modelio skaičiavimų skaičius nepriklauso nuo modelio parametrų skaičiaus, taip pat nepriklauso nuo modelio rezultato skirstinio tipo ir jo savybių, tačiau visi modelio parametrai tarpusavyje turi būti nepriklausomi.

Wilks metodas numato, kad (α, β) tolerancijos intervalui apskaičiuoti reikalingas minimalus imties tūris n turi tenkinti (1.18) nelygybę, (α, β) tolerancijos ribos atveju – (1.19) nelygybę:

$$1 - \alpha^n - n(1 - \alpha)\alpha^{n-1} \geq \beta \quad (1.18)$$

$$1 - \alpha^n \geq \beta \quad (1.19)$$

Pagal (1.18) ir (1.19) formules apskaičiuoti reikalingi minimalūs tūriai dažniausiai naudojamiems α ir β reikšmėms pateikti 1.1-1.2 lentelėse. Kiekiai gauti pagal (1.18) ir (1.19) formules rodo, kad reikia 64 skaičiavimų, kad sudaryti imtį, kuria pakankamai tiksliai ($\beta=0,99$) apskaičiuotume tolerancijos intervalą apimantį ne mažiau 90% ($\alpha=0,9$) visų modelio rezultatų. Tačiau jei norėtume mažiau tiksliai ($\beta=0,90$) apskaičiuoti tolerancijos intervalą, bet apimantį ne mažiau 99% ($\alpha=0,99$) visų modelio rezultatų, reikėtų net 388 skaičiavimų. Praktiniuose skaičiavimuose dažniausiai taikomas $(0,95; 0,95)$ - tolerancijos intervalas yra pakankamai geras kompromisas tarp tikslumo ir reikalingo modeliavimų skaičiaus.

1.1 lentelė

Minimalus skaičiavimų kiekis
 (α, β) tolerancijos intervalui

$\alpha \backslash \beta$	0,90	0,95	0,99
0,90	38	77	388
0,95	46	93	473
0,99	64	130	662

1.2 lentelė

Minimalus skaičiavimų kiekis
 (α, β) tolerancijos ribai

$\alpha \backslash \beta$	0,90	0,95	0,99
0,90	22	45	230
0,95	29	59	299
0,99	44	130	459

1.2.4. JAUTRUMO ANALIZĖS TAIKYMAI

Rezultatų jautrumo analizės paskirtis yra dvejopa: pirma – nustatyti sistemos modeliavimo rezultatų jautrumą nuo pradinių parametrų, antra – analizuoti pagrindinių modeliavimo prielaidų svarbą galutiniams rezultatams. Dažniausiai naudojama kiekybinė neapibrėžto rezultato jautrumo

analizė [11] – tai parametų kategorizavimas pagal jų santykinę indėlį visam rezultato neapibrėžtumui ir kiekybinis šio indėlio įvertinimas kiekvienam parametru.

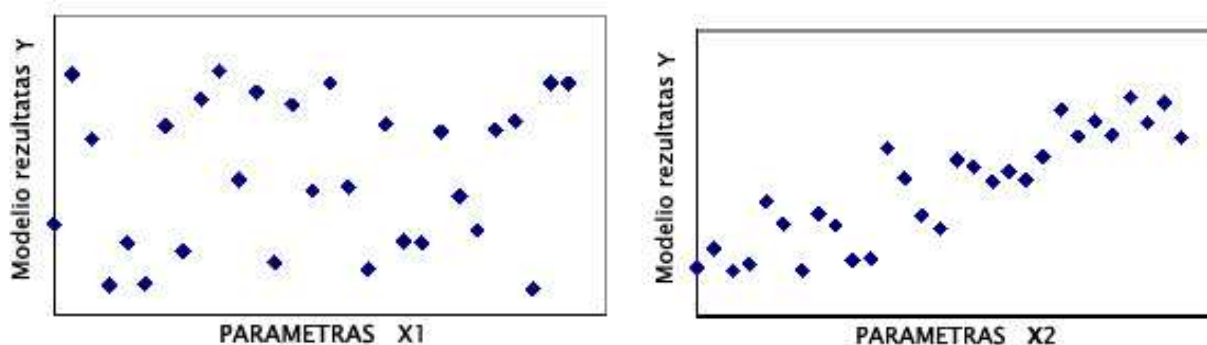
Vienas pagrindinių jautrumo analizės tikslų yra modelio rezultatų kitimo įvertinimas, kai keičiami modelio parametrai. Toks vertinimas yra svarbus siekiant:

- įvertinti modelio taikymo galimybes;
- nustatyti tuos parametrus, kurie yra reikšmingiausi norint gauti kuo tikslesnius modelio rezultatus;
- suprasti pagrindines modeliuojamos sistemos funkcionavimo priklausomybes.

Pagal jautrumo analizės rezultatus galime nustatyti, kurių modelio parametų tikslesnis įvertinimas leistų ženkliai sumažinti modelio rezultato neapibrėžtumą, ir kurių parametų tolimesnis tikslinimas nėra prasmingas dėl jų mažos įtakos rezultato kitimui. Kadangi parametų įvertinimas dažnai susijęs su turimomis žiniomis apie tam tikrus fizikinius dydžius ar reiškinius, parametro tikslesnis įvertinimas gali reikalauti atlikti papildomus eksperimentinius tyrimus. Todėl jautrumo analizė gali padėti numatyti prioritetinius eksperimentinius tyrimus ir tuo pačiu objektyviai spręsti optimalaus lėšų paskirstymo brangiems tyrimams klausimą.

1.2.5. JAUTRUMO ANALIZĖS METODAI

Vienas paprasčiausių jautrumo analizės metodų yra sklaidos grafikų analizė. Paprastai rekomenduojama atlikti tokią grafinę analizę jautrumo analizės pradiniam etape, kuris leistų nesunkiai pastebėti esamus tiesioginius sąryšius tarp modelio parametų ir rezultato. Sklaidos grafikams gauti naudojama ta pati parametų ir modelio rezultatų imtis, kaip ir neapibrėžtumo analizei atlikti. Sklaidos grafiko pavyzdžiai pateikti 1.10 paveiksluose. Pavyzdžiuose matyti, kad parametro X_1 reikšmės nėra tiesiškai susijusios su modelio rezultato Y reikšmėmis, tačiau parametro X_2 atveju tokia priklausomybė egzistuoja. Kuo parametro X_2 reikšmės yra mažesnės, tuo mažesnės ir modelio rezultato reikšmės, o kuo parametro X_2 reikšmės didesnės – tuo ir didesnės rezultato reikšmės. Toliau pateiksime metodus, leidžiančius kiekybiškai įvertinti šią priklausomybę.



1.10 pav. Sklaidos grafikų pavyzdžiai

Vienas paprasčiausių apskaičiuojamų parametų jautrumo indeksų yra koreliacijos koeficientas. Praktikoje skaičiuojami tiek Pirsono koreliacijos koeficientas (1.21), tiek ir ranginis Spirmeno koreliacijos koeficientas [6,7].

$$R_i = \frac{1/n \sum_j (x_{i,j} - Ex_i)(y_j - Ey)}{\sqrt{\left(n \sum_j x_{i,j}^2 - \left(\sum_j x_{i,j} \right)^2 \right) \left(n \sum_j y_j^2 - \left(\sum_j y_j \right)^2 \right)}}, \quad (1.20)$$

čia $x_{i,j}$ žymi i -ąją sistemos parametro reikšmę atliekant j -ąją modeliavimą; Ex_i - x_i parametro vidurkis; Ey - modelio vidurkis; $i = 1, 2, \dots, k$; $j = 1, 2, \dots, n$; k - parametų kiekis, n - imties tūris.

Parametro x_i koreliacijos koeficientas nusako tiesinę priklausomybę tarp parametro ir modelio rezultato. Kuo absoliutiniu dydžiu koreliacijos reikšmė arčiau vieneto, tuo parametro x_i įtaka modelio rezultatui didesnė. Tokiu būdu galima reitinguoti modelio parametrus pagal jų įtaką modelio rezultatui. Pagrindiniai koreliacijos koeficiento kaip jautrumo indekso trūkumai yra tokie: koreliacijos koeficientas turi prasmę tik esant tiesinei priklausomybei ir nėra vertinama keleto parametų tarpusavio sąveikos įtaka modelio rezultatui. Jeigu parametų ir modelio rezultato priklausomybė nėra tiesinė, o taip dažnai ir būna, galima pabandyti eliminuoti netiesiškumo įtaką ranguojant duomenis ir skaičiuojant koreliacijos koeficientą ne duomenims, o jų rangams. Tada parametų jautrumo indeksą atitinka ranginis Spirmeno koreliacijos koeficientas, kurio interpretacija ir trūkumai nagrinėjant rangus yra analogiški Pirsono koreliacijos koeficientui. Nei vienas iš koreliacijos koeficientų negali įvertinti keleto parametų tarpusavio sąveikos įtakos, todėl koreliacijos koeficientai yra tinkami parametų jautrumo indeksai tik, jei parametų tarpusavio sąveikos įtaka modelio rezultatui yra nežymi.

Literatūroje [7,11,16] galima sutikti ir dalinės koreliacijos koeficiento taikymus parametų jautrumo analizėje. Dalinė koreliacija atskleidžia rezultato y priklausomybę nuo vieno kintamojo, kai eliminuojama likusių kintamųjų įtaka. Jis yra apibūdinamas kaip koreliacijos koeficientas tarp $y - \hat{y}$ ir $x_i - \hat{x}_i$, $i = 1, 2, \dots, k$, čia \hat{y} ir \hat{x}_i išreiškiami regresijos lygtimis: $\hat{y} = b_0 + \sum_{j \neq i} b_j x_j$ ir $\hat{x}_i = c_0 + \sum_{j \neq i} c_j x_j$

Vienas populiariausių imties metodų jautrumo analizėje yra standartizuota tiesinė regresija. Šiuo atveju matematinis modelis (1.17) išreikšimas daugialype tiesine parametų funkcija:

$$y = F(x_1, x_2, \dots, x_N) = a + b_1 \cdot x_1 + \dots + b_N \cdot x_N \quad (1.21)$$

Daugialypės regresijos koeficientai b_i apskaičiuojami mažiausių kvadratų metodu, tačiau jie negali būti jautrumo indeksais, kadangi dėl skirtingų parametų matavimo vienetų parametų išsibarstymo ribos gali būti labai įvairios. Parametų jautrumą galima palyginti tik tada kai yra lyginama vienodai galinčius kisti dydžius, todėl atliekamas parametų ir rezultatų sunormavimas:

$$\hat{x}_{i,k} = \frac{x_{i,k} - Ex_i}{\sigma x_i}, i = 1, 2, \dots, N; k = 1, 2, \dots, M$$

$$\hat{y}_k = \frac{y_k - Ey}{\sigma y}$$

- čia Ex_i - x_i parametro vidurkis;
 Ey – modelio rezultato vidurkis;
 σx_i - x_i parametro standartinis nuokrypis;
 σy - modelio rezultato standartinis nuokrypis;
 M – parametrų atsitiktinės imties dydis;
 N - parametrų skaičius.

Tuomet regresijos koeficientai β_i standartizuotiems dydžiams vadinami standartizuotais regresijos koeficientais (SRK) ir yra dažnai naudojami parametrų jautrumo indeksai:

$$\hat{y} = \alpha + \beta_1 \cdot \hat{x}_1 + \dots + \beta_N \cdot \hat{x}_N \quad (1.22)$$

Viena iš svarbiausių prielaidų, kuomet SRK turi jautrumo indekso prasmę, yra prielaida kad tiesinio modelio (1.21), (1.22) determinacijos koeficientas R^2 artimas vienetui. Jautrumo analizėje determinacijos koeficiento reikšmė nurodo tą tiesinio modelio rezultato neapibrėžtumo dalį, kurią galima paaiškinti modelio parametrų neapibrėžtumu. Likusi nepaaiškinama modelio rezultato neapibrėžtumo dalis susijusi su modelio parametrų tarpusavio sąveikos įtaka rezultatui ir ji gali būti analizuojama dispersijos išskaidymo metodais [17]. Praktikoje dažnai reikalaujama, kad tiesinio modelio determinacijos koeficientas būtų ne mažesnis, nei 0,6, t.y. parametrų neapibrėžtumai paaiškintų ne mažiau, nei 60% modelio rezultato neapibrėžtumo. Jei R^2 yra mažesnis, tuomet SRK pateikiamas parametrų jautrumo reitingavimas gali būti klaidingas. Esant mažam R^2 , koreliacijos koeficientai taip pat negali būti parametrų jautrumo indeksais.

Jei jautrumo analizei atlikti imties metodais naudojama maža imtis (taip dažniausiai būna naudojant Wilks metodą), tuomet tarp tam tikrų modelio parametrų gali susidaryti taip vadinama momentinė koreliacija. Ši koreliacija atsiranda tik dėl to, kad esant mažai imčiai neįmanoma užtikrinti, kad ji tenkintų atsitiktinės imties savybes. Momentinė koreliacija gali iškreipti modelio parametrų jautrumo indeksus ir pateikti klaidingą jų reitingavimą. Todėl rekomenduojama palyginti kelių imties metodų pateikiamus rezultatus (SRK, koreliacijos koeficientus). Jei jautrumo reitingavimas tam tikriems parametrų ženkliai skiriasi, reikėtų ištirti parametrų reikšmių imties savybes.

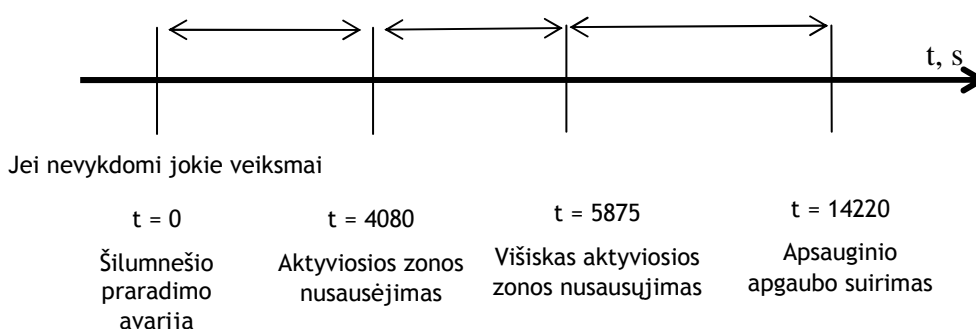
2. TIRIAMOJI DALIS

2.1. TIKIMYBINĖS DINAMIKOS PATIKIMUMO ANALIZĖ

2.1.1. TIKIMYBINĖS DINAMIKOS SISTEMA

Atominė elektrinė (AE) aprūpiną regioną sąlyginai pigia elektros energija. Kartu AE yra ypač didelio pavojaus šaltinis. Sutrikus elektrinės veiklai išskyla pavojus tūkstančių žmonių gyvybėms, taip pat milijonai žmonių gali būti apšvitinti radioaktyviais spinduliais, kurie sukelia įvairias lėtines ligas. Todėl procesų vykstančių AE saugumas yra svarbiausias prioritetas, kuriam turi būti nuolat skiriamas dėmesys. Viena iš aktualiausių AE saugumo problemų yra branduoliniame reaktoriuje galinti įvykti šilumnešio praradimo avarija. Dėl aušinamojo skysčio praradimo reaktoriuje gali susidaryti sąlygos, kurios sukelia vandenilio sproginimą. Savo ruožtu, sproginimas gali pažeisti apsauginį apgaubą, saugantį aplinką nuo radioaktyviųjų spindulių. Darbo tikslas įvertinti apsauginio apgaubo pažeidimo tikimybę, jo pasiskirstymo tankį laike bei nustatyti įvykių scenarijus, kuriems įvykstant apgaubo pažeidimo tikimybė yra didžiausia.

Šiame darbe nagrinėjamas uždavinys yra sudarytas pagal prancūzų 900 MWe suslėgto vandens reaktorių su pasyviaisiais autokatalitiniais rekombaineriais. Laikoma, kad šilumnešio praradimo avarija įvyksta pradiniu laiko momentu $t=0$. Nustatyta, kad po 1 val. ir 8 min.(4080 s.) prasideda reaktoriaus aktyviosios zonos nusausėjimas. Jeigu nesiimama jokių saugumo priemonių po 3 val. 57 min (14220 s.) įvyksta apsauginio apgaubo suirimas. Kadangi iki aktyviosios zonos nusausėjimo situacija laikoma saugia ir iki tol nesiimama jokių saugumo priemonių, todėl avarija nagrinėjama nuo laiko momento $t=4080s$.



2.1 pav. Šilumnešio praradimo avarijos poveikis

Branduolinio reaktoriaus sistema apibūdinama šiais kintamaisiais: vandenilio, vandens garų, deguonies ir azoto koncentracija moliais, bendra vandenilio mase, kuri išsiskyrė dėl kuro kanalų

dangos oksidacijos, slėgiu bei temperatūra. Laikoma, kad slėgis ir temperatūra po apsauginiu apgaubu visur yra vienodi, o dujų mišinys yra homogeninis. Kol nėra aktyvuojama jokia saugos sistema, visi kintamieji išskyrus vandenilio koncentraciją išlieka pastovūs. Pradinės parametrų reikšmės pateiktos 2.1 lentelėje.

2.1 Lentelė
Sistemos parametrai ir jų pradinės reikšmės

Vandenilio kiekis, n_{H_2}	0 mol.
Vandens garų kiekis, n_{H_2O}	1717196 mol.
Deguonies kiekis, n_{O_2}	368384 mol.
Azoto kiekis, n_{N_2}	1390372 mol.
Bendra vandenilio masė išsiskyrusi dėl kuro kanalų dangos oksidacijos, M_{H_2}	0 kg.
Temperatūra, T	382,05 K.
Slėgis, p	2,34 bar.

Tuo tarpu, įvykus aktyviosios zonos nusausėjimui vandenilis išsiskiria ir patenka į apsauginį apgaubą tam tikru srautu q_{H_2} , kuris kinta priklausomai nuo laiko momento ir yra nusakytas 2.2 lentelėje

2.2 Lentelė
Išsiskiriančio vandenilio masės srautas

Laikas t , s	Srautas q_{H_2} , kg/s
4081	0
5340	0,00066
5660	0,028
5970	0,17
6110	0,36
6425	0,156
7005	0,04
7410	0,09
9180	0,034
11070	0

Branduolinis reaktorius turi kelias apsaugos sistemas:

- **Pasyvieji autokatalitiniais rekombaineriai (PAR)** perdirba vandenilį ir deguonį į vandenį, taip sumažinama vandenilio koncentracija. Perdirbto vandenilio srautas $q_{H_2}^{par}$ apskaičiuojamas pagal (2.1) formulę.

$$q_{H_2}^{par} = (3p + 3.7) \frac{100n_{H_2}}{N} \quad (2.1)$$

čia n_{H_2} yra vandenilio molekulių kiekis, N – bendras molekulių kiekis, p – slėgis.

Perdirbto deguonies srautas yra aštuonis kartus didesnis už vandenilio. PAR veikia nuolat, nepriklausomai nuo sistemos būklės.

- **Vandens užliejimo sistema** (SIS, angl. Safety Injection System) aktyviają zoną užlieja vandeniu 7,5 kg/s srautu. Jeigu SIS prasideda prieš visišką aktyviosios zonos nusausėjimą ($t=5875s.$), laikoma, kad situacija tampa saugia, tai yra, vandenilio sprogimai nebegalimi, apgaubo pažeidimas neįvyks. Tikimybė, kad vandens užliejimas įvyks prieš visišką aktyviosios zonos nusausėjimą yra 0,5. Sistemos aktyvacijos laiko momentas yra pasiskirstęs pagal tolygųjį skirstinį.

Jeigu SIS nebuvo įjungtas iki $t=5875s.$, yra 0,5 tikimybė, kad vandens užliejimas įvyks iki apgaubo suirimo. Tokiu atveju, SIS įjungimo laiko momentas taip pat pasiskirstęs pagal tolygųjį skirstinį. Jeigu SIS įjungiamas po visiško aktyviosios zonos nusausėjimo, visas vandens srautas dalyvauja reakcijoje su plieniu ir cirkoniu, išsiskirdamas 0,833 kilogramus vandenilio per sekundę ($q_{SIS}=0,833 kg/s$). Maksimali išsiskyrusio vandenilio masė M_{H_2} yra 950 kg. Pasiekus šią ribą laikoma, kad vandens užliejimas sustoja.

Tikimybė, kad vandens užliejimas nebus visiškai įjungtas yra lygi 0,25.

2.3 Lentelė

Vandens garų koncentracijos kitimas įjungus apgaubo atšaldymo sistemą

Nr., i	Laikas t_i , s	Vandens garų kiekis n_i , mol.
0	T_0 s (CHRS įjungimas)	1717196
1	T_0+600	968499
2	T_0+1500	416979
3	T_0+2700	232292
4	T_0+8000	201399

2.4 Lentelė

Temperatūros kitimas įjungus apgaubo atšaldymo sistemą

Laikas t , s	Vidutinė temperatūra T , K
T_0 s (CHRS įjungimas)	382
T_0+600	364
T_0+1500	346
T_0+2700	334
T_0+8000	328
$T_0+13000$	323

- **Apgaubo atšaldymo sistema** (CHRS, angl. Containment Heat Removal System) purškia vandenį į apgaubą, taip kondensuodama vandens garus ir sumažindama temperatūrą bei slėgį viduje. Tikimybė, kad atšaldymo sistema bus aktyvuota lygi 0,5, o jos įjungimo laiko momentas yra pasiskirstęs pagal tolygųjį skirstinį.

Aktyvavus CHRŠ, vandens garų koncentracija keičiasi pagal dėsnį, kuris pateiktas 2.3 lentelėje, temperatūros kritimas apibūdinamas 2.4 lentelėje pateiktais duomenimis..

Nepaisant apsaugos sistemų, susidarius palankioms sąlygoms apgaubo viduje gali susidaryti sproginiai. Sprogimas įmanomas, tik tada jeigu yra tenkinama sąlyga:

$$[H_2] \geq [H_2]_{degi}, \quad (2.2)$$

čia $[H_2]$ yra vandenilio koncentracija, $[H_2]_{degi}$ yra ribinė degi vandenilio koncentracija, kuri yra išreiškiama per garų koncentraciją $[H_2O]$:

$$[H_2]_{degi} = 0.0399 + 0.0402[H_2O] + 0.1625[H_2O]^2. \quad (2.3)$$

Dėl itin aukštos temperatūros rekombineriai gali pagreitinti sproginio atsiradimą, jeigu yra tenkinama labai sproginio koncentracijos sąlyga:

$$[H_2] \geq [H_2]_{sprogi}, \quad (2.4)$$

čia

$$[H_2]_{sprogi} = 0.052 + 0.0964[H_2O]. \quad (2.5)$$

Net jeigu tenkinamos (2.3) ir (2.5) sąlygos, sproginis atsiranda po tam tikro uždelsimo, kurio trukmė priklauso nuo reaktoriaus viduje esančios vandenilio koncentracijos $[H_2]$:

- jeigu $[H_2] = [H_2]_{degi}$, uždelsimas iki sproginio yra tolygiai pasiskirstęs tarp 0 ir 4 valandų;
- jeigu $[H_2]_{degi} < [H_2] < [H_2]_{sprogi}$, uždelsimas yra tolygiai pasiskirstęs tarp 0 ir D_{Tmax} valandų,

$$DT_{max} = \frac{4([H_2] - [H_2]_{sprogi}) - 0.333([H_2] - [H_2]_{degi})}{[H_2]_{sprogi} - [H_2]_{degi}} \quad (2.6)$$

- jeigu $[H_2] \geq [H_2]_{ignis}$, uždelsimas yra tolygiai pasiskirstęs tarp 0 ir 0,333 valandų.

Laikoma, kad sproginis įvyksta akimirksniu. Jo metu sudegusio vandenilio dalis yra pasiskirsčiusi tarp 0,05 ir 1 pagal tolygų skirstinį. Sproginio metu susidaręs pikinis slėgis yra paskaičiuojamas pagal specialią procedūrą, kurią pateikė Lietuvos energetikos institutas.

Tiesioginis sproginio poveikis sistemai yra:

- sumažėjęs vandenilio molekulių kiekis;
- sumažėjęs deguonies molekulių kiekis;
- padidėjęs vandens garų kiekis;
- laikinai padidėjęs slėgis.

Sproginio metu susidaręs pavojingai didelis pikinis slėgis gali sukelti apsauginio apgaubo pažeidimą. Laikoma, kad apgaubas yra pažeidžiamas, jeigu slėgis viduje tampa didesnis už 5 barų ribą.

Apibūdinta sistema atitinka dinaminės sistemos charakteristikas. Daugelis aprašytų sistemos procesų yra dinaminiai, todėl negalime nepaisyti laiko faktoriaus ir šios sistemos patikimumo vertinti įprastais gedimų ir įvykių medžiais. Nors saugos sistemos įsijungia be vėlavimų ir dauguma procesų pasikeitimų įvyksta akimirksniu, tačiau pats svarbiausias įvykis – vandenilio sproginimas turi atsitiktinį uždelsimą, kurio negalime nepaisyti. Todėl aprašytą dinaminę sistemą reikia vertinti, kaip stimuliuojamos dinamikos atvejį ir taikyti 1.1.5 skyriuje pateiktą teoriją.

2.1.2. SISTEMOS FORMALIZAVIMAS

Norit modeliuoti aprašytą tikimybinę dinamiką reikia ją formalizuoti, tai yra, identifikuoti įvykius, dinamikas, stimulus ir uždelsimus bei sistemos kintamuosius. Taip pat nustatyti jų tarpusavio sąryšius ir priklausomybes.

Įvykiai

2.1.1 skyriuje apibrėžtoje dinaminėje sistemoje, galimi 5 skirtingi įvykiai įtakojantys sistemos elgseną ir būseną:

1. vandens užliejimo sistemos (SIS) įjungimas;
2. vandens užliejimo sistemos (SIS) išjungimas;
3. apgaubo atšaldymo sistemos (CHRS) įjungimas;
4. vandenilio sproginimo atsiradimas;
5. apsauginio apgaubo pažeidimo atsiradimas.

Priklausomai nuo apsaugos sistemos SIS įjungimo laiko momento, ji gali turėti skirtingą poveikį sistemai. Jeigu vandens užliejimas įjungiamas prieš visišką aktyviosios zonos nusausėjimą ($t=5875s.$), jis stabilizuoja situaciją, laikoma, kad sistema yra saugioje būsenoje. Tačiau, jeigu SIS įjungimo laiko momentas yra vėlesnis negu $t=5875s.$, išsiskiria itin daug vandenilio ir sistemos gali tapti pavojinga.

2.5 Lentelė

Sistemos įvykiai

Nr.	Žymėjimas	Aprašymas
0	SIS0	SIS įjungimas iki visiško aktyviosios zonos nusausėjimo
1	SIS1	SIS įjungimas po visiško aktyviosios zonos nusausėjimo
2	SIS*	SIS išjungimas
3	CHRS	Apgaubo atšaldymo sistemos įjungimas
4	sprog	Vandenilio sproginimo atsiradimas
5	pažeid	Apsauginio apgaubo pažeidimo atsiradimas

Kad vienareikšmiškai atskirti įvykio poveikį, sudaromi du atskiri įvykiai:

- SIS įjungimas iki visiško aktyviosios zonos nusausėjimo;
- SIS įjungimas po visiško aktyviosios zonos nusausėjimo.

Visi aprašyti įvykiai apibendrintai pateikti 2.5 lentelėje.

Dinamikos

Pagal atsitikusius ir neatsitikusius įvykius sudaromos 7 skirtingos dinamikos, kurios pateiktos 2.6 lentelėje.

2.6 Lentelė

Sistemos dinamikos

Nr.	SISO	SIS1	CHRS	sprog	pažeid
0	0	0	0	0	0
1	1	\	\	\	\
2	0	1	0	0	0
3	0	0	1	0	0
4	0	1	1	0	0
5	\	\	\	1	0
6	\	\	\	\	1

Stimulai ir uždelsimai

Pagal stimuliuojamos dinamikos teoriją kiekvieną įvykį sukelia aktyvuotas stimulus, kuris gali būti apibrėžiamas kaip tam tikros sąlygos tenkinimas arba nekontroliuojamas atsitikimas. Po stimulo aktyvacijos turi praeiti tam tikras uždelsimas iki pačio įvykio atsiradimo. Uždelsimas taip pat gali būti apibūdintas tikimybinio skirstiniu arba determinuota sąlyga. Uždelsimo metu sistema kinta pagal esamą dinamiką, todėl per tą laiką pasikeitus sistemos sąlygomis ir joms tapusiomis nepalankios įvykio atsiradimui, turi įvykti stimulo deaktyvacija.

Pagal pateiktą avarijos aprašymą yra nustatyti 6 sistemos dinamikas įtakojantys įvykiai. 3 iš jų yra saugos sistemų įjungimai. Pagal pateiktas prielaidas laikoma, kad saugos sistemos įjungiamos atsitiktinai, jos pradeda veikti pilna jėga be jokio vėlavimo ar uždelsimo. Todėl formaliai saugos sistemos įjungimą galime aprašyti, kaip stimulo su nuliniu uždelsimu aktyvaciją. Šių stimulų aktyvacija apibrėžiama tikimybinio tankiu, kuris nustatomas kiekvienai saugos sistemai atskirai.

Tuo tarpu vandens užliejimo išjungimas ir apsauginio apgaubo pažeidimas yra apibrėžiami determinuotomis sąlygomis, todėl jų stimulai yra nusakomi Dirako funkcijomis.

Ypatingą dėmesį reikia skirti sprogimo įvykiui. Sprogimo stimulo aktyvacija apibrėžiamas determinuota sąlyga – vandenilio koncentracija pasiekia sprogią ribą. Sprogimo uždelsimas yra išreiškiamas tikimybiniais tankiais, kurie priklauso nuo esamos koncentracijos. Sprogimas taip pat turi stimulo deaktyvaciją. Jeigu vandenilio koncentracija sumažėja, sprogimas nebeįmanomas, todėl aktyvuoto stimulo uždelsimas turi būti nutrauktas.

Visų įvykių stimulų tikimybinės charakteristikos pateiktos 2.7 lentelėje.

2.7 Lentelė

Įvykių stimulų ir uždelsimų charakteristikos

Įvykis	Stimulo aktyvacija	Uždelsimas	Stimulo deaktyvacija	Deaktyvacijos uždelsimas
SIS0	$\frac{0,5}{5875 - 4080}$ jeigu $t < 5875$	$\delta(t')$	/	/
SIS1	$\frac{0,5}{14220 - 5875}$ jeigu $5875 \leq t \leq 14220$	$\delta(t')$	/	/
SIS*	$\delta(M_{H_2} - 950)$	$\delta(t')$	/	/
CHRS	$\frac{0,5}{14220 - 4080}$ jeigu $4080 \leq t \leq 14220$	$\delta(t')$	/	/
sprog	$\delta([H_2] - [H_2]_{deg i})$	$\frac{1}{14400}$ jei $[H_2] = [H_2]_{deg i}$ $\frac{1}{DT_{max}}$ jei $[H_2] > [H_2]_{deg i}$ $[H_2] < [H_2]_{sprog i}$ $\frac{1}{1200}$ jei $[H_2] \geq [H_2]_{sprog i}$	$\delta([H_2]_{deg i} - [H_2])$	$\delta(t^*)$
pažeid	$\delta(p - 5)$	$\delta(t')$	/	/

Sistemos kintamieji

Kaip pateikta 2.1 lentelėje sistema apibūdinama 7 kintamaisiais. Toliau apibrėšime, kaip jie kinta priklausomai nuo to, kurioje dinamikoje sistema yra:

- Vandenilio molekulių kiekis n_{H_2} kinta pagal (2.7) dėsnį:

$$\frac{dn_{H_2}}{dt} = \frac{q}{m_m(H_2)}, \quad (2.7)$$

q – suminis išsiskiriančio ir perdirbamo vandenilio masės srautas, kuris pateikiamas 2.8 lentelėje, $m_m(H_2)$ – vandenilio molinė masė.

2.8 Lentelė

Suminis vandenilio masės srautas

Dinamika	Suminis vandenilio masės srautas q , kg/s
Dinamika 0	$q_{H_2} - q_{par}$
Dinamika 1	$saugi\ būsena (\backslash)$
Dinamika 2	$q_{H_2} - q_{par} + q_{sis}$
Dinamika 3	$q_{H_2} - q_{par}$
Dinamika 4	$q_{H_2} - q_{par} + q_{sis}$
Dinamika 5	sprogimas
Dinamika 6	\backslash

čia q_{par} perdirbamo vandenilio srautas, paskaičiuotas pagal (2.1) formulę, $q_{SIS} = 0.833\text{kg/s}$, q_{H2} apibūdinamas 2.2 lentele, išsiskiriantis vandenilio srautas tarpiniuose laiko momentuose apskaičiuojamas panaudojus tiesinę interpoliaciją.

- *Vandens garų kiekis* n_{H2O} priklauso nuo to, kiek laiko veikia apgaubo atšaldymo sistema (CHRS) bei įvykusių sproгимų. Kadangi sproгимai gali padidinti vandens garų kiekį ir pakeisti sistemoje esančias sąlygas, negalime skaičiuoti garų kiekio kitimą atsižvelgiant tik į laiko faktorių. Todėl pasinaudodami (2.8) formule pagal 2.3 lentelėje pateiktus duomenis galime nustatyti, kaip garų kondensacijos greitis priklauso nuo sistemoje esančių garų kiekio.

$$\frac{dn_{H2O}}{dt} = \frac{n_i - n_{i-1}}{t_i - t_{i-1}} \quad (2.8)$$

čia i žymi numerį vandens garų kiekio iš 2.3 lentelės, kuris tenkina sąlygą:

$$n_{i-1} \leq n_{H2O} < n_i \quad (2.9)$$

Apibendrintai, kaip vandens garų kiekis apskaičiuojamas priklausomai nuo esančios dinamikos, pateikta 2.9 lentelėje.

2.9 Lentelė

Vandens garų kiekio skaičiavimas priklausomai nuo esamos dinamikos

Dinamika	Vandens garų kiekio kitimas
Dinamika 0	garų kiekis nekinta
Dinamika 1	saugi būseną (\)
Dinamika 2	garų kiekis nekinta
Dinamika 3	(2.8) lygtis
Dinamika 4	(2.8) lygtis
Dinamika 5	sproгимas
Dinamika 6	\

- *Deguonies molekulių kiekis* n_{O2} keičiasi tik dėl veikiančių pasyviųjų autokatalitinių rekombainerių. Deguonies kiekio n_{O2} kitimas aprašomas (2.10) dėsnium, kuris išlieka toks pats nepriklausomai nuo esamos dinamikos, išskyrus sproгимo atvejį.

$$\frac{dn_{O2}}{dt} = -\frac{8 \cdot q_{H2}^{par}}{m_m(O_2)}, \quad (2.10)$$

čia $m_m(O_2)$ yra deguonies molinė masė.

- *Azoto molekulių kiekis* n_{N2} yra konstanta esant visoms dinamikos. Sproгимas jų kiekio taip pat neįtakoja, todėl toliau šio kintamojo nenagrinėsime.

- *Bendra vandenilio masė išsiskyrusi dėl kuro kanalų dangos oksidacijos* M_{H2} paskaičiuojamas pagal (2.11) lygtį:

$$\frac{dM_{H_2}}{dt} = q_{MH_2}, \quad (2.11)$$

čia q_{MH_2} yra bendras išsiskiriančio vandenilio srautas, kuris priklausomai nuo dinamikos yra apibūdintas 2.10 lentelėje.

2.10

Lentelė Bendras išsiskiriančio vandenilio srautas

Dinamika	Bendras išsiskiriančio vandenilio srautas, q_{MH_2}
Dinamika 0	q_{H_2}
Dinamika 1	saugi būseną (\)
Dinamika 2	$q_{H_2} + q_{SIS}$
Dinamika 3	q_{H_2}
Dinamika 4	$q_{H_2} + q_{SIS}$
Dinamika 5	sprogimas
Dinamika 6	\

- *Temperatūros T* kitimą apibūdina 2.11 lentelė.

2.11 Lentelė

Temperatūros kitimas priklausomai nuo esamos dinamikos

Dinamika	Temperatūros kitimas
Dinamika 0	temperatūra nekinta
Dinamika 1	saugi būseną (/)
Dinamika 2	pagal 2.4 lentelę
Dinamika 3	temperatūra nekinta
Dinamika 4	pagal 2.4 lentelę
Dinamika 5	temperatūra nekinta
Dinamika 6	/

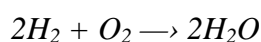
- *Slėgis p* yra paskaičiuojamas remiantis formule:

$$p = 2,34 \frac{N}{3475\,952}, \quad (2.12)$$

čia N žymi bendrą molių kiekį esantį sistemoje.

Vandenilio sproginimas

Vandenilio sproginimas vyksta pagal gerai žinomą cheminę reakciją:



Vandenilio, deguonies ir vandens garų molių kiekių priklauso nuo to, kokia vandenilio dalis B sudegė. Sprogimo metu įvykę cheminiai pasikeitimai apibūdamai (2.13) lygtimis:

$$\begin{aligned} n_{H_2}^+ &= (1 - B) \cdot n_{H_2}^- \\ n_{O_2}^+ &= \left(1 - \frac{1}{2} \cdot B\right) \cdot n_{O_2}^- \end{aligned} \quad (2.13)$$

$$n_{H_2O}^+ = (I + B) \cdot n_{H_2O}^-$$

čia „-“ žymi elemento molių kiekį prieš sproginimą, „+“ – elemento molių kiekis po sproginimo.

Sproginimo metu susidaręs slėgio pikas apskaičiuojamas pagal AICC funkciją, kurią pateikė Lietuvos energetikos institutas.

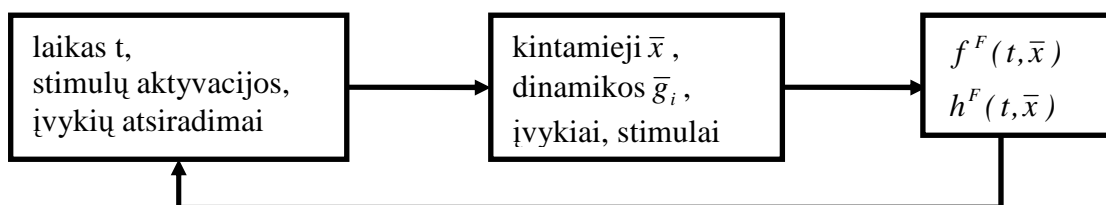
2.1.3. IMITACINIO MODELIAVIMO METODOLOGIJA

1.1.5 skyriuje pateikta stimuliuojamos dinamikos teorija pateikia analitines išraiškas (1.4)-(1.7) kaip įvertinti nepageidaujamo būvio atsiradimo tikimybę. Tačiau pateikta metodologija yra per daug sudėtinga ir kompleksiška, kad ją būtų galima taikyti praktikoje. Todėl pateikto uždavinio sprendimui sudaromas imitacinis modelis, kuris nesudėtingai gali būti pritaikomas kitai stimuliuojamai dinamikai modeliuoti.

Modeliuojant įvairias sudėtingas matematinės ir fizikinės sistemas itin plačiai yra taikomas Monte Karlo imitacinis metodas. Tai yra skaitmeninis algoritmas, kurio veikimas remiasi atsitiktiniais (arba pseudo-atsitiktiniais) skaičiais, o galutinis rezultatas nustatomas iš klaidų ir bandymo santykio. Svarbiausia ir sudėtingiausia metodo dalis – sudaryti tikslų sistemos modelį. Tam reikia nustatyti sistemos veikimo principus, numatyti galimas situacijas, kurių metu keičiasi dinamikos ar atsiranda pažeidimai nutraukiantys tolimesnį procesą.

Sudarydami stimuliuojamos tikimybinės dinamikos sistemos veikimo imitacinį modelį galime išskirti 3 svarbius elementus:

- **Laikas** - nepriklausomas faktorius kintantis pastoviu žingsniu. Kaip ir realiam gyvenime, laikas yra kontroliuojantis elementas. Jo kitimo metu keičiasi sistemos parametrų reikšmės pagal nustatytą deterministinę dėsnį arba atsiranda įvykiai, stimulų aktyvacijos.
- **Deterministinės charakteristikos**, kurias sudaro kintamieji, įvykiai, stimulai, dinamikos. Deterministinės charakteristikos apibūdina sistemos momentinę sistemos būseną, tai yra, nusako kintamųjų reikšmės \bar{x} ir dėsnį pagal kurį jie kinta. Akivaizdu, kad sistemos būseną yra priklausoma nuo laiko $\bar{x} = \bar{g}_i(\bar{u}, t)$.
- **Tikimybinėmis charakteristikomis** nusakoma stimulų aktyvacijos ir įvykių atsiradimo tankio funkcijos. Nustatant ryšį su kitais elementais galima pastebėti, kad tikimybinėms charakteristikoms didelę reikšmę turi esama sistemos būseną. Tik žinant sistemos parametrų reikšmės ir aktyvią dinamiką, galima paskaičiuoti įvykio tankį duotu laiko momentu.



2.2 pav. Vidiniai modelio sąryšiai tarp tikimybinių ir deterministinių charakteristikų

Kaip pavaizduota 2.2 paveiksle kiekvienas iš šių elementų siejasi vienas su kitu ir daro įtaką. Ši schema yra svarbi tuo, kad realizuojant programinėmis priemonėmis galima modelį išskaidyti į atskiras dalis. Tai suprastina programą ir kartu padaro ją lankstesnę ir lengviau pritaikomą kitiems uždaviniais. Schema taip pat atskleidžia modelio vidinius sąryšius tarp determinuotų ir stochastinių procesų. Remiantis sudaryta schema galima sudaryti tokius programinius modulius:

- centrinis valdymo modulis;
- determinuotų procesų modulis;
- tikimybinių procesų modulis.

Centrinis valdymo modulis atliekant daugkartinį modeliavimą bei fiksuoja kiek kartų sistema buvo pažeista. Kiekvieną laiko momentą jis inicijuoja sistemos būsenos įvertinimą, nusprendžia kurie stimulai yra aktyvuojami, ar atsiranda kokie nors įvykiai. Determinuotų procesų modulis yra sudarytas iš dinamikų ir įvykių realizavimo, esamos būklės analizės ir besivystančio scenarijaus fiksavimu bei saugojimu. Tikimybinių procesų modulis priklausomai nuo besivystančio scenarijaus paskaičiuoja stimulų aktyvacijos, uždelsimų pasibaigimo, įvykių atsiradimo tikimybes tam tikrame laiko intervale.

Prieš analizuojant sudėtingą aprašytą uždavinį, sudarykime paprastą stimuliuojamos dinaminės sistemos, kuri yra sudaryta iš vieno stimulo F ir vieno galimo įvykio, imitavimo modelį. Tarkime, kad yra sistema kintanti pagal nustatytą determinuotą dėsnį (dinamiką) $\bar{x} = \bar{g}_1(\bar{u}, t)$, $\bar{u} = \bar{x}|_{t=t_0}$ tol kol neatsiranda įvykis, kurį sukelia aktyvuotas stimulus F . Tokios sistemos veikimą galima apibūdinti paprastu algoritmu:

1. sistemos parametrai kinta pagal deterministinę dėsnį $\bar{x} = \bar{g}_1(\bar{u}, t)$, $t > t_0$;
2. atsitiktiniu laiko momentu τ_F aktyvuojamas stimulus F ;
3. sistemos parametrai toliau kinta pagal deterministinę dėsnį $\bar{x} = \bar{g}_1(\bar{u}, t)$, kol vyksta laiko uždelsimas τ iki įvykio atsiradimo. Laiko uždelsimas – atsitiktinis dydis;
4. laiko momentu $t_F = \tau_F + \tau$ atsiranda įvykis.

Tai yra labai paprastas algoritmas, kuris atitinka stimulo aprašymą pateiktą 1.1.5 skyriuje. Kartu šiame algoritme puikiai atskleisti vidiniai sistemos sąryšiai, kuriuos iliustruoja 2.2 paveikslas. Kaip matome, laiko faktorius yra nepriklausomas ir valdantis determinuotą sistemos kitimą elementas, tačiau kad jis galėtų sukelti įvykį yra būtini atsitiktiniai dydžiai: stimulo F aktyvacijos laiko momentas τ_F ir laiko uždelsimas τ . Šie dydžiai yra charakterizuojami tikimybiniais tankiais $f^F(t)$ ir $h^F(t)$.

Nagrinęjant realias dinamines sistemas, kuriose sistemos parametrai gali kisti pagal vieną iš kelių determinuotų dėsnų (dinamikų), stimulų aktyvacijos tankio funkcijos $f_i^F(t, \bar{x})$ ir uždelsimo tankio funkcijos $h_i^F(t, \bar{x})$ yra žinomos kiekvienai dinamikai i atskirai, be to, jos yra priklausomos nuo sistemos parametrų reikšmių \bar{x} . Kaip matome, modelio tikimybiniai ir deterministiniai elementai yra labai glaudžiai vienas su kitu susiję, tai neleidžia iš anksto numatyti, kada įvyks stimulo F aktyvacija

bei kiek laiko truks uždelsimas. Todėl stimulo aktyvacijos laiko momentas τ_F generuojamas tokiu būdu:

- sugeneruojamas atsitiktinis dydis p pasiskirstęs pagal tolygųjį skirstinį intervale $[0; 1]$;
- nustatoma $s(t_0) = 0$;
- laikoma, kad laikas t kinta pradedant t_0 žingsniu Δt , $t_i = t_{i-1} + \Delta t$;
- laiko momentais t_i atliekami tokie veiksmai:
 - paskaičiuojamos sistemos parametrų reikšmės \bar{x} ;
 - pagal aktyvią dinamiką ir gautas \bar{x} reikšmes nustatoma, kokia tikimybė p_i , kad stimulus bus aktyvuotas per laiko intervalą $[t_{i-1}; t_i]$;
 - $s(t_i) = s(t_{i-1}) + p_i$;
 - jeigu $s(t_i) \geq p$, tuomet t_i yra stimulo aktyvacijos laiko momentas.

Šis algoritmas leidžia generuoti laiko momentus nepaisant to, kaip pasikeičia stimulo aktyvacijos tankio funkcija priklausomai nuo besikeičiančių dinamikų ir sistemos parametro reikšmių. Šį algoritmą taip pat galima taikyti generuojant laiko uždelsimą τ , kuris trunka iki įvykio atsiradimo.

Analizuojamą paprastą vieno stimulo modelį praplėtus iki bendro modelio su keliomis dinamikomis, įvykiais, stimulais ir uždelsimais gautume tokį stimuliuojamos dinamikos imitacinio modelio algoritmą:

- stimulų aktyvacijų ribinių tikimybių nustatymas
- sistemos iniciavimas
- pažingsnis laiko keitimas, pradedant t_0 , baigiant t_n
- kiekvienu laiko momentu t_i vykdomi tokie veiksmai
 - kreipiasi į sistemos modulį, kad pasiskaičiuotų parametrų reikšmes
 - kiekvienam neaktyvuotam stimului paskaičiuojama aktyvacijos kaupiamoji tikimybė
 - jeigu ji yra lygi arba didesnė už sugeneruotą ribinę tikimybę
 - aktyvuojamas stimulus
 - inicijuojamas įvykio atsiradimo uždelsimas
 - atsitiktinai nustatoma ribinė uždelsimo pasibaigimo tikybė
 - nustatoma nulinė kaupiamoji uždelsimo tikimybė
 - kiekvienam aktyvuotam stimului paskaičiuojama įvykio uždelsimo kaupiamoji tikimybė
 - jeigu ji yra lygi arba didesnė už sugeneruotą ribinę tikimybę
 - inicijuojamas atsiradęs įvykis
 - nustatoma, ar atsiradęs įvykis nepažeidė sistemos
 - nutraukiama stimulo aktyvacija
 - atsitiktinai nustatoma nauja stimulo aktyvacijos ribinė tikimybė
 - nustatoma nulinė kaupiamoji stimulo aktyvacijos tikimybė
 - jeigu įvyko sistemos pažeidimas
 - jis yra fiksuojamas
 - pradedamas sistemos veikimas imitavimas iš pradžių

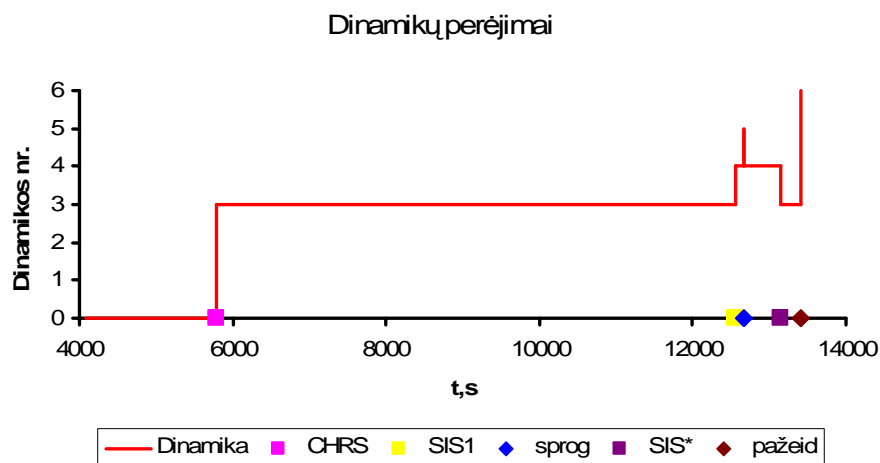
Kartu šis algoritmas sudaro Centrinio valdymo modelio pagrindą ir esmę. Jame realizuotas pilnas sistemos valdymas: sistemos kintamųjų paskaičiavimas, stimulų aktyvavimas, uždelsimo pasibaigimo fiksavimas. Kaip matome, šis algoritmas yra nepriklausomas nuo sistemos modelio ir gali būti

pritaikytas bet kokiai stimuliuojamai dinamikai, kurios deterministinės ir tikimybinės charakteristikos realizuotos atskirai savo moduluose.

2.1.4. SISTEMOS TIKIMYBINĖ SAUGOS ANALIZĖ

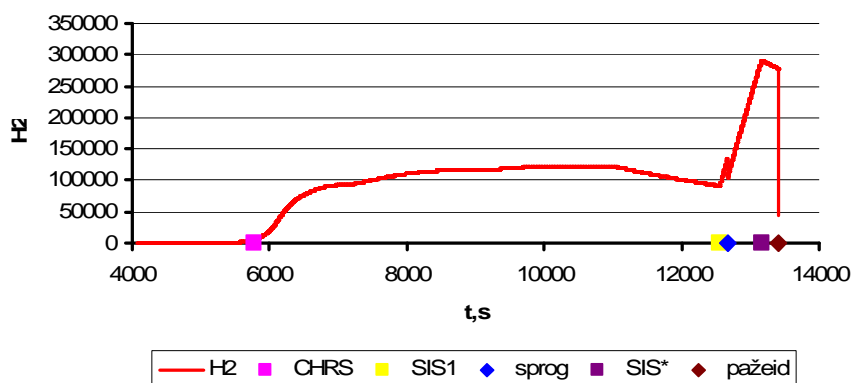
Pagal sudarytą tikimybinės dinamikos imitacinio modeliavimo metodologiją buvo sukurtos programinės priemonės leidžiančios modeliuoti 2.1.2 skyriuje formalizuota sistema. Prieš atliekant sistemos patikimumo analizę, paanalizuosime kaip kinta sistemos kintamieji vienos imitacijos metu.

Sistemos imitavimas prasideda laiko momentu $t=4080$ esant nulinėje dinamikoje (2.3 paveikslas). Pirmoji įsijungusi saugos sistema yra apgaubo atšaldymo sistema (CHRS), kuri aktyvuojama laiko momentu $t=5784$. Vandens garų molių kiekis pradeda greitai kondensuotis (2.5 paveikslas), todėl krenta slėgis apgaubo viduje (2.6 paveikslas). Laiko momentu $t=12550$ sistema pereina iš trečios dinamikos į ketvirtą, kadangi tuo metu įsijungia vandens užliejimo sistema (SIS). Dėl plieno ir cirkonio oksidacijos sparčiai pradeda kilti vandenilio koncentracija (2.4 paveikslas), dėl to netrukus ($t=12662$) įvyksta pirmas sproginimas. Atsiradęs sproginimas nestiprus, jo metu susidaręs pikinis slėgis nepasiekia 5 barų ribos, todėl laikoma, kad apgaubo pažeidimas neįvyksta. Tačiau kol veikia SIS, vandenilio molių kiekis sparčiai didėja. Laiko momentu $t=13159$ oksiduoja visi plienas ir cirkonis, dėl to laikoma, kad vandens užliejimas sustoja ir sistema grįžta į 3 dinamiką. Dėl veikiančių pasyviųjų autokatalitinių rekombainerių pradeda mažėti vandenilio molių kiekis. Nepaisant to įvyksta antrasis sproginimas. Kaip matome, jo metu sudega didžioji dalis vandenilio, sproginimo metu susidaręs slėgis beveik pasiekia 6 barus, dėl to pažeidžiamas apsauginis apgaubas.



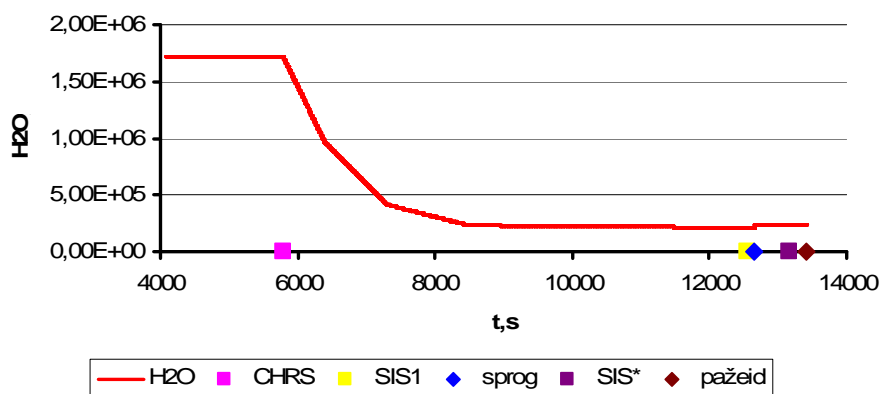
2.3 pav. Dinamikų perėjimai

Vandenilio kiekio kitimo grafikas



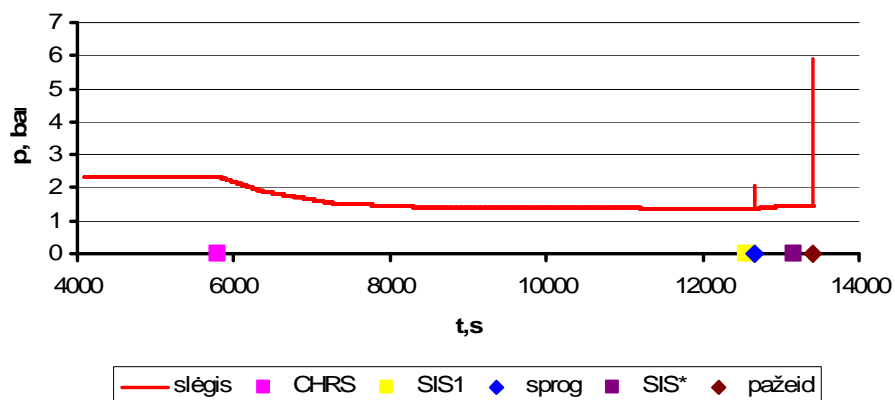
2.4 pav. Vandenilio molių kiekio kitimo grafikas

Vandens garų kiekio kitimo grafikas



2.5 pav. Vandens garų molių kiekio kitimo grafikas

Slėgio kitimo grafikas



2.6 pav. Slėgio kitimo grafikas

Kaip matome, sukurta programine priemone galima puikiai imituoti 2.1.1 skyriuje aprašytą dinaminę sistemą. Ja pasinaudojus buvo atlikta tikimybinė saugos analizė.

Atliekant saugos analizę svarbiausia yra atsakyti į šiuos klausimus:

1. Kaip dažnai gali įvykti pavojingas įvykis?
2. Kas turi atsitikti, kad jis atsitiktų?
3. Kokios pasekmės yra numatomos, įvykiui atsitikus?

Analizuojant mūsų sistemą reikia nustatyti:

1. kokia tikimybė, kad atsitiks apsauginio apgaubo pažeidimas,
2. kokios saugos sistemos turi suveikti ar nesuveikti, kad atsitiktų apsauginio apgaubo sužeidimai;
3. kokie galimi nuostoliai įvykus pažeidimui.

Kadangi mūsų nagrinėjamo uždavinio tikslas yra patikimumo analizė, šiame darbe trečio klausimo nenagrinėsime ir koncentruosimės ties pirmais dviem.

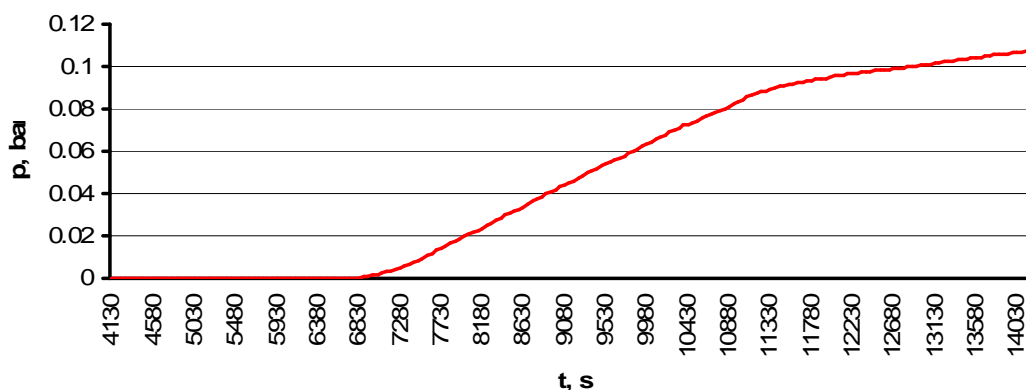
Pirmiausiai įvertinsime tikimybę, kad įvyks apsauginio apgaubo pažeidimas. Šiam tikslui buvo atlikta 100 000 imitacinių sistemos modeliavimų taikant vienos sekundės laiko kitimo žingsnį. Kaip pateikta 2.12 lentelėje apgaubo pažeidimas įvyko 10787 kartų, gautas apgaubo pažeidimo tikimybės įvertis lygus 0,10787. Kiekvieną sistemos modeliavimą laikant Bernulio eksperimentu, buvo įvertintas pažeidimo tikimybės pasikliautinis intervalas. Kaip matome, su pasikliovimo lygmeniu 0,95 pažeidimo tikimybės pasiklautinis intervalas yra (0,1059; 0,1098)

2.12 Lentelė

Sistemos modeliavimo rezultatai

Imitavimų kiekis:	100000
Pažeidimų kiekis:	10787
Apgaubo pažeidimo tikimybės įvertis (pažeidimo dažnis):	0,10787
Apgaubo pažeidimo tikimybės pasikliautinis intervalas su pasikliovimo lygmeniu 0,95:	(0,1059, 0,1098)

Apsauginio apgaubo pažeidimo dažnio kitimas laike



2.7 pav. Apsauginio apgaubo pažeidimo dažnio kitimas laike

2.7 paveikslas vaizduoja kaip kinta sistemos pažeidimo dažnis laike. Galime pastebėti, kad pažeidimo tikimybės įvertis lygus nuliui iki laiko momento $t=6800s$. Tai gali būti paaiškinta tuo, kad kol nėra visiškai nusausėjus aktyvioji zona ($t=5875s$.) išsiskyrusio vandenilio masės srautas yra labai mažas dėl to nesusidaro palankios sąlygos sproginams. Tačiau kuo toliau, tuo labiau sistemoje esantis dujų mišinys tampa pavojingas, todėl pažeidimo dažnis tolygiai didėja. Nuo $t=6000s$. iki $t=11000s$. funkcija įgyja tiesinę formą ir galime pastebėti, kad būtent šiame laiko intervale yra didžiausia tikimybė, kad įvyks pažeidimas. Nuo laiko momento $t=11000s$. pažeidimo dažnis auga nebe taip greitai.

Toliau atliksime scenarijų analizę, tai yra, nustatysime, kuriuos saugos sistemos turi suveikti ar nesuveikti, kad atsirastų sistemos pažeidimas. Pagal tai, ar suveikia vandens užliejimas ir apgaubo atšaldymo sistema darbo metu buvo sudaryti 6 scenarijai, kurie pateikti 2.13 lentelėje. Taip pat buvo apskaičiuota kiekvieno scenarijaus atsiradimo tikimybė. Teorinė tikimybė buvo įvertinta neatsižvelgiant į tai, kad vandenilio sproginas gali nutraukti tam tikrus scenarijus po pirmo atsiradusio įvykio ir taip įtakoti tų scenarijų pasirodymo dažnį.

2.13 Lentelė

Apsaugos sistemų suveikimo scenarijai

Scenarijus	Pirmas įvykis	Antras įvykis	Scenarijaus atsiradimo tikimybė
Scenarijus A	\	\	0,125
Scenarijus B	CHRS	\	0,125
Scenarijus C	CHRS	SIS	0,096
Scenarijus D	SIS	CHRS	0,051
Scenarijus E1	SIS0	\	0,478
Scenarijus E2	SIS1	\	0,125

Atlikus imitacinį sistemos modeliavimą buvo nustatyta, kad sproginas daro įtaką scenarijų atsiradimams. Dalis sistemos modeliavimų buvo nutraukti po pirmo atsitikusio įvykio, todėl tam tikrų scenarijų pasirodymo tikimybės sumažėjo, o kitų padidėjo. Gauti scenarijų atsiradimo tikimybių įverčiai pateikti 2.14 lentelėje.

2.14 Lentelė

Scenarijų atsiradimo tikimybės

Scenarijus	Gautas tikimybės įvertis	Teorinė tikimybė	Santykinė paklaida, %
A	0,12112	0,125	3,1
B	0,12361	0,125	1,1
C	0,09652	0,096	0,5
D	0,02748	0,051	46,1
E1	0,48311	0,478	1,1
E2	0,14816	0,125	18,5

Kaip matome, A, B, C ir E1 scenarijų pasirodymo tikimybių įverčiai beveik sutampa su teorinėmis reikšmėmis, jų santykinė paklaida neviršija 5 proc. Tuo tarpu, E2 scenarijaus tikimybės įverčio ir teorinės tikimybės skirtumas sudaro 18,5 proc. santykinę paklaidą. Taip pat matome, kad scenarijaus D įvertis yra dvigubai mažesnis už teorinę tikimybę. Tai atsitiko dėl to, kad dalis modeliavimų, kuriuose po vandens užliejimo turėjo būti įjungta apgaubo atšaldymo sistema buvo nutraukti taip ir nespėjus aktyvuoti antros saugos sistemos. Todėl ir taip maža D scenarijaus pasirodymo tikimybė dar labiau sumažėja.

Kad būtų galima daryti išvadas apie kiekvieno scenarijaus pavojingumą sistemai, imitacinio modeliavimo metu buvo nustatyta:

- kiek kartų buvo realizuotas scenarijus;
- scenarijaus metu įvykusių pažeidimų kiekis;
- scenarijaus metu įvykusių sproгимų kiekis;
- tikimybės, kad įvyks apsauginio apgaubo pažeidimas, jeigu realizuojamas duotas scenarijus, įvertis;
- sproгимų dažnio įvertis;
- vidutinis sproгимų metu susidaręs pikinis slėgis;
- sproгимų metu susidariusių pikinių slėgių dispersija.

Kiekvieno scenarijaus visos apibrėžtos charakteristikos pateiktos 2.15 lentelėje.

2.15 Lentelė

Scenarijų tikimybinės charakteristikos

Scenarijus	Realizacijų kiekis	Pažeidimų keikis	Sproгимų kiekis	Sąlyg. pažeidimo tikimybės įv.	Sproгимų dažnio įvertis	Pikinio slėgio vidurkus, bar	Pikinio slėgio dispersija, bar
A	12112	0	0	0	0	0	0
B	12361	0	1799	0	0,145538	2,56553	0,370876
C	9652	3111	11643	0,322317	1,206278	4,01378	2,2792
D	2748	1001	3563	0,364265	1,296579	4,43525	1,99252
E1	48311	0	0	0	0	0	0
E2	14816	6675	9057	0,450526	0,611299	6,08772	2,48864

Didžiausią sąlyginę pažeidimo tikimybę turi E2 scenarijus. Tai reiškia, įjungus vandens užliejimo sistemą po aktyviosios zonos nusausėjimo, apgaubo pažeidimas įvyks su tikimybe 0,45. Kaip matome, šio scenarijaus metu susidarę sproгимai nėra tokie dažni kaip C ir D scenarijuose, tačiau vidutiniškai E2 sproгимai yra stipriausi ir tai lemia dažną apgaubo pažeidimą. Taip pat galime pastebėti, kad C ir D scenarijų sproгимo dažnių, pikinio slėgio empirinės charakteristikos yra labai panašios, tačiau C scenarijus įvyksta tris kartus dažniau.

Kadangi C, D ir E2 scenarijus sieja tai, kad juose visuose yra aktyvuojama vandens užliejimo sistema po aktyviosios zonos nusausėjimo, apibendrintai galima pasakyti, kad vandens užliejimas daro neigiamą poveikį sistemos patikimumui. Kaip rodo atlikti skaičiavimai, šią sistemą galima įjungti tik iki aktyviosios zonos nusausėjimo, vėliau ji sukelia nepageidaujamų pasekmių, kurių išvengti negalima net įjungus kitą saugos sistemą.

2.2. NEAPIBRĖŽTUMO IR JAUTRUMO ANALIZĖ

2.2.1. DUOMENŲ IR SISTEMOS MODELIO NEAPIBRĖŽTUMAI

Kaip ir daugumos patikimumo analizės modelių, 2.1.1 skyriuje aprašytos sistemos parametrai buvo nustatyti pagal statistinius duomenis, parenkant labiausiai tikėtiną reikšmę. Kad galėtume įvertinti gautų modelio rezultatų taikymo galimybes bei nustatyti reikšmingiausius parametrus būtina atlikti neapibrėžtumo ir jautrumo analizę. Pirmas neapibrėžtumo analizės žingsnis yra sistemos duomenų bei parametru neapibrėžtumų identifikavimas:

- Sistemos kintamųjų pradinės reikšmės yra išsibarsčiusios pagal tolygųjį skirstinį tarp minimalios ir maksimalios reikšmių, kurios pateiktos 2.16 lentelėje.

2.16 Lentelė
Sistemos kintamųjų pradinių reikšmių išsibarstymo ribos

Kintamasis	Minimali reikšmė	Maksimali reikšmė
Vandenilio kiekis, n_{H_2}	0 mol.	0 mol.
Vandens garų kiekis, n_{H_2O}	1383240 mol.	2034004 mol.
Degunies kiekis, n_{O_2}	368384 mol.	
Azoto kiekis, n_{N_2}	1390372 mol.	
Bendra vandenilio masė išsiskyrusi dėl kuro kanalų dangos oksidacijos, M_{H_2}	0 kg.	0 kg.
Temperatūra, T	369,75 K	383,55 K
Slėgis, p	2 bar.	2,5 bar.

- Nustatoma, kad vandens užliejimo sistemos (SIS) tiekiamo vandens srauto stiprumas priklauso nuo esamo laiko momento. Dėl įjungtos SIS bei įvykstančios plieno ir cirkonio oksidacijos, išsiskiriančio vandenilio masės srauto neapibrėžtumas pateikiamas trijuose skirtinguose laiko intervaluose (2.17 lentelė). Pavyzdžiui, laiko intervale nuo $t=5875s.$ iki $t=7000s.$ dėl oksidacijos išsiskiriančio vandenilio masės srautas gali būti lygus $0,333kg/s.$ su tikimybe $0,3$; $0,416kg/s.$ su tikimybe $0,4$; $0,5kg/s.$ su tikimybe $0,3$. Laikoma, kad laiko išsiskiriančio vandenilio srautas išlieka pastovus vieno intervalo ribose.

2.17 Lentelė

**Dėl plieno ir cirkonio oksidacijos išsiskiriančio
vandenilio masės srauto neapibrėžtumai**

Laiko intervalas	Vandenilio masės srautas (tikimybė)	Vandenilio masės srautas (tikimybė)	Vandenilio masės srautas (tikimybė)
[5875s; 7000s]	0,333 kg/s (0,3)	0,416 kg/s (0,4)	0,5 kg/s (0,3)
[7001s; 9400s]	tolygiai pasiskirstęs tarp 0,416 kg/s ir 0,833 kg/s		
[9401s; 14220s]	0,333 kg/s (0,3)	0,416 kg/s (0,4)	0,5 kg/s (0,3)

Tikimybė, kad vandens užliejimo sistemą bus galima įjungti po aktyviosios zonos nusausėjimo yra pasiskirsčiusi pagal diskretų skirstinį:

SIS1 aktyvacijos dažnis	0,3	0,4	0,5	0,6	0,7
SIS1 aktyvacijos dažnio tikimybė	0,2	0,2	0,3	0,2	0,1

- Kol apgaubo atšaldymo sistema (CHRS) nėra aktyvuota, vandens garų molių kiekis ir temperatūra po apsauginiu apgaubu nekinta. Įjungus CHRS vandens garų kiekis keičiasi pagal 2.18 lentelėje pateiktus duomenis.

2.18 Lentelė

Vandens garų kiekio kitimo neapibrėžtumai

Laikas, s	Vandens garų kiekis, mol.
T_0 s (CHRS aktyvacijos momentas)	1717196
$T_0+600 \cdot E_{CHRS}$	968499
$T_0+1500 \cdot E_{CHRS}$	416979
$T_0+2700 \cdot E_{CHRS}$	232292
$T_0+8000 \cdot E_{CHRS}$	201399

čia E_{CHRS} yra CHRS sistemos efektyvumo koeficientas tolygiai pasiskirstęs tarp 0,5 ir 1,5. E_{CHRS} nekinta sistemos modeliavimo metu. Vandens garų kiekiui tarpiniams laiko momentams paskaičiuoti naudojama tiesinė interpoliacija.

Taip pat yra pateikiamas apgaubo atšaldymo sistemos aktyvacijos tikimybės diskretusis pasiskirstymas:

CHRS aktyvacijos dažnis	0,3	0,4	0,5	0,6	0,7
CHRS aktyvacijos dažnio tikimybė	0,2	0,2	0,3	0,2	0,1

- Pasyvieji autokatalitiniai rekombaineriai (PAR) funkcionuoja sistemoje visą laikotarpį nepaisant jos būklės. Perdirbto vandenilio masės srautas apskaičiuojamas pagal (2.14) formulę.

$$q_{H_2}^{par} = (K1 \cdot p + K2) \frac{100n_{H_2}}{N} \quad (2.14)$$

čia n_{H_2} yra vandenilio molekulių kiekis, N – bendras molekulių kiekis, p – slėgis. Parametrai K_1 ir K_2 paskaičiuojami pagal formules

$$\begin{aligned} K_1 &= 3 \cdot E_{K1} \\ K_2 &= 3,7 \cdot E_{K2} \end{aligned} \quad (2.15)$$

čia E_{K1} ir E_{K2} yra nepriklausomi PAR efektyvumo koeficientai tolygiai pasiskirstę tarp 0,5 ir 1,5.

- Sprogimo atsiradimo sąlygos išlieka tokios pačios kaip ir pirminiame sistemos modelyje. Tačiau laikoma, kad sprogimo metu sudegusio vandenilio kiekis yra tolygiai pasiskirstęs tarp F_{min} ir 1, F_{min} paskaičiuojamas pagal (2.16) formulę:

$$F_{min} = 0,05 + ([H_2] - 0,04) \frac{0,9}{6} \quad (2.16)$$

čia $[H_2]$ yra vandenilio koncentracija sistemoje.

- Lyginant su pirminiu sistemos modeliu apibrėžtu 2.1.1 skyriuje keičiasi apgaubo pažeidimo sąlyga. Apgaubo pažeidimo tikimybė priklauso nuo sprogimo metu susidariusio slėgio ir yra apibrėžta 2.19 lentelėje. Jeigu sprogimo metu susidarė tarpinis slėgis, apgaubo pažeidimo tikimybė paskaičiuojama tiesiškai interpoliuojant 2.19 lentelėje esančias tikimybes.

2.19 Lentelė

Pažeidimo dažnio neapibrėžtumas

Sprogimo metu susidaręs slėgis, bar.	Apgaubo pažeidimo tikimybė
3	$0,00 \cdot E_{pažeid}$
4	$0,02 \cdot E_{pažeid}$
5	$0,05 \cdot E_{pažeid}$
6	$0,10 \cdot E_{pažeid}$
7	$0,50 \cdot E_{pažeid}$
8	$0,75 \cdot E_{pažeid}$
9	$0,90 \cdot E_{pažeid}$
10	$1,00 \cdot E_{pažeid}$

čia $E_{pažeid}$ yra pažeidimo ribos neapibrėžtumo koeficientas. $E_{pažeid}$ yra tolygiai pasiskirstęs tarp 0,5 ir 1,5.

Apibendrintai visų parametru neapibrėžtumai pateikiami 2.20 lentelėje.

2.20 Lentelė

Sistemos duomenų ir parametru neapibrėžtumai

Nr.	Žymėjimas	Pavadinimas	Neapibrėžtumas
1.	$E_{n_{H_2O}}$	Pradinis vandens garų kiekis	T(1383240 mol.; 2034004 mol.)
2.	E_T	Pradinė temperatūra	T(369,75 K.; 383,55 K.)
3.	E_p	Pradinis slėgis	T(2 bar.; 2,5 bar.)

Nr.	Žymėjimas	Pavadinimas	Neapibrėžtumas
4.	E_{K1}	PAR pirmas efektyvumo koeficientas	T(0,5; 1,5)
5.	E_{K2}	PAR antras efektyvumo koeficientas	T(0,5; 1,5)
6.	E_{SIS_1}	SIS metu išsiskiriančio vandenilio srautas laiko intervale [5875s; 7000s]	D(0,333 kg/s (0,3); 0,416 kg/s (0,4); 0,5 kg/s (0,3))
7.	E_{SIS_2}	SIS metu išsiskiriančio vandenilio srautas laiko intervale [7001s; 9400s]	T(0,416 kg/s; 0,833 kg/s)
8.	E_{SIS_3}	SIS metu išsiskiriančio vandenilio srautas laiko intervale [9401s; 14220s]	D(0,333 kg/s (0,3); 0,416 kg/s (0,4); 0,5 kg/s (0,3))
9.	E_{SIS_A}	SIS aktyvacijos po visiško aktyviosios zonos nusausėjimo tikimybė	D(0,3 (0,2); 0,4 (0,2); 0,5 (0,3); 0,6 (0,2); 0,7 (0,1))
10.	E_{CHRS}	CHRS efektyvumo koeficientas	T(0,5; 1,5)
11.	E_{CHRS_A}	CHRS aktyvacijos tikimybė	D(0,3 (0,2); 0,4 (0,2); 0,5 (0,3); 0,6 (0,2); 0,7 (0,1))
12.	$E_{pažeid}$	Pažeidimo ribos neapibrėžtumo koeficientas.	T(0,5; 1,5)

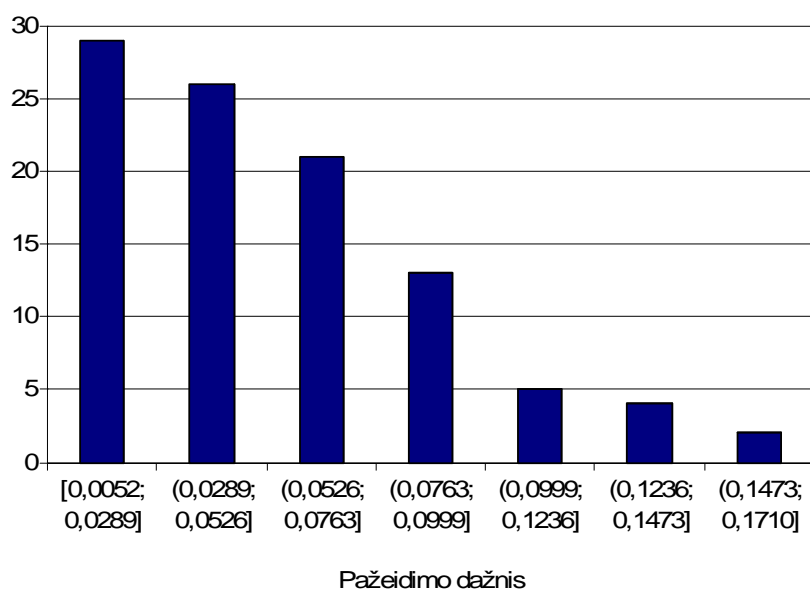
čia PAR – pasyvieji autokatalitiniai rekombaineriai; SIS – vandens užliejimo sistema; CHRS – apgaubo atšaldymo sistema; T(a, b) – tolygusis skirstinys intervale (a, b); D (a (p_a)) – diskretusis skirstinys, a – reikšmė, p_a – reikšmės tikimybė

2.2.2. MODELIO NEAPIBRĖŽTUMO ANALIZĖ

Sudaryto modelio neapibrėžtumo analizė buvo atlikta apgaubo pažeidimo tikimybės įverčio atžvilgiu. Kadangi dėl ilgos sistemos modeliavimo trukmės neįmanoma sudaryti labai didelės rezultatų imties, buvo atsitiktinai parinkta 100 pradinių duomenų rinkinių, kurį sudaro 2.20 lentelėje pateikti parametrai bei pradiniai duomenys. Su kiekvienu duomenų rinkiniu buvo atliktas sistemos modeliavimas, kiekvieną kartą atliekant 10 tūkstančių sistemos imitavimų. Pagal gautus rezultatus įvertinamas apgaubo pažeidimo dažnio neapibrėžtumas bei jautrumas atskiriems modelio parametrams.

Visų pirma buvo nustatyta apgaubo pažeidimo dažnio histograma (2.8 paveikslas) bei paskaičiuoti rezultatų skaitinių charakteristikų įverčiai (2.21 lentelė). Iš gautų rezultatų matyti, kad apgaubo pažeidimo dažnis yra išsivarsęs plačiame intervale, nuo 0,0052 iki 0,171, įvertinus modelio parametru neapibrėžtumą, labiausiai tikėtinas apgaubo pažeidimo dažnis lygus intervalo [0,0052; 0,0289] vidurinei reikšmei 0,017. 2.1 skyriuje gautas pažeidimo tikimybės įvertis nepatenka į labiausiai tikėtiną intervalą ir yra dvigubai didesnis už vidutinį pažeidimo dažnį. Tačiau dėl to negalime atmesti galimybės, kad sistemos pažeidimo tikimybė gali siekti ir viršyti 0,10787 ribą.

Apgaubo pažeidimo dažnio histograma



2.8 pav. Apgaubo pažeidimo dažnių histograma

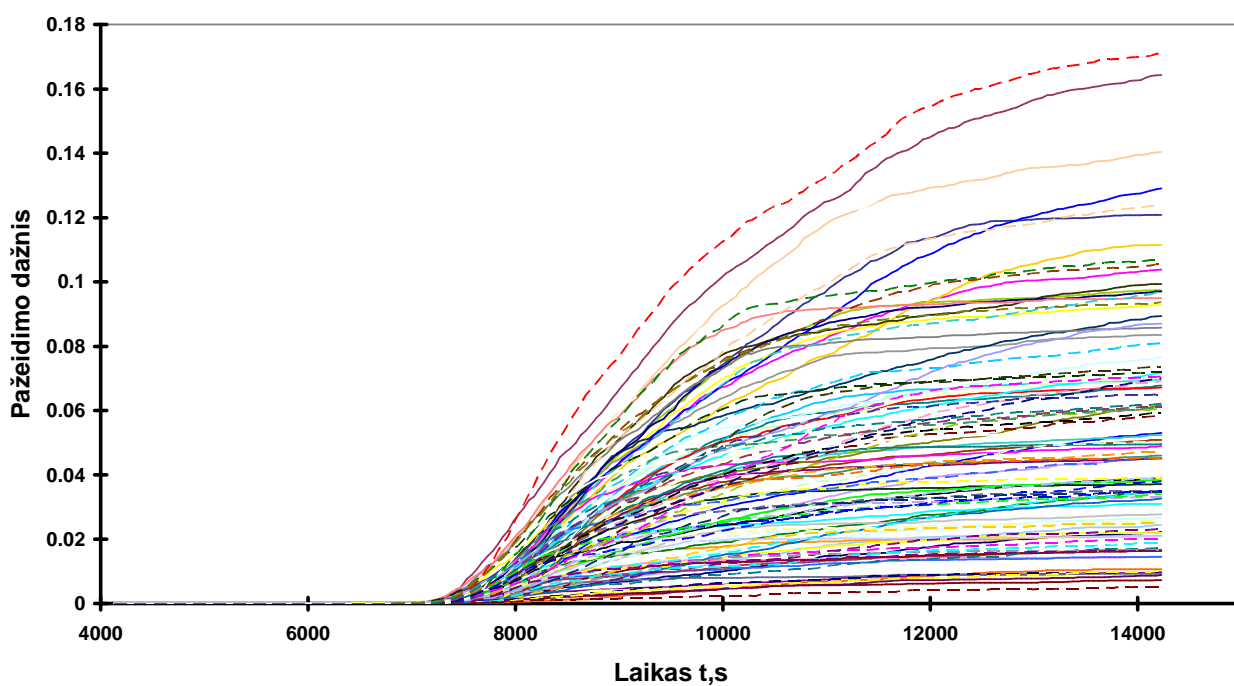
2.21 Lentelė

Apgaubo pažeidimo dažnio empirinės charakteristikos

Vidurkis	0,055209
Standartinis nuokrypis	0,036949
Mediana	0,0471
Moda	0,0170
Minimumas	0,0052
Maksimumas	0,1710
(0,95, 0,95) Tolerancijos intervalas	(0,0052; 0,171)
(0,95, 0,95) Tolerancijos intervalo viršutinė riba	0,16430

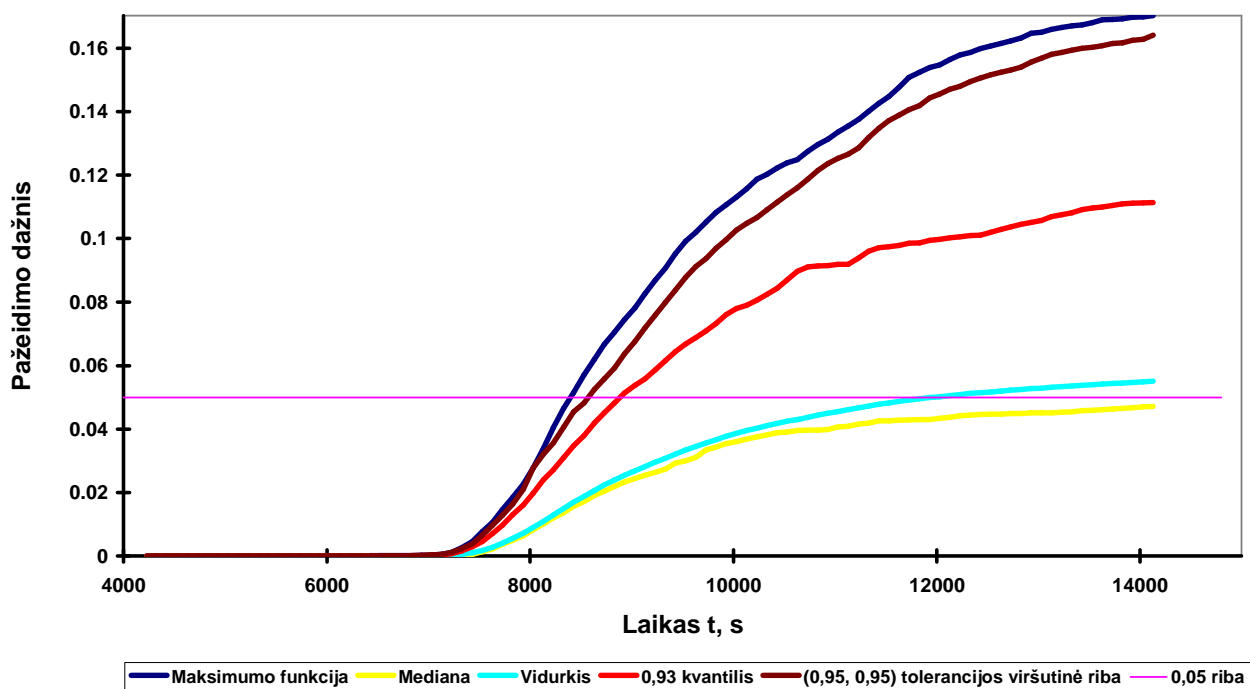
Sudarytos rezultatų imties tūris yra 100, tokios imties užtenka, kad galėtume įvertinti (0,95, 0,95) tolerancijos intervalą. Pagal gautus modeliavimo rezultatus, pažeidimo dažnio minimumą ir maksimumą atitinka reikšmės 0,0052 ir 0,171, todėl su tikimybe 0,95, 95 procentai visų modeliavimo rezultatų patenka į (0,0052, 0,171) intervalą. Taip pat paskaičiuota (0,95, 0,95) tolerancijos intervalo viršutinė riba rodo, kad su tikimybe 0,95 95 procentai modeliavimo rezultatų neviršys 0,16430 apgaubo pažeidimo tikimybės ribos. Šie skaičiavimai rodo, kad įvertinus modeliavimo parametrų ir pradinių duomenų neapibrėžtumus, su tikimybe 0,95 sumodeliuota apgaubo pažeidimo tikimybė 95 atvejais iš šimto neviršys 0,16430 ribos arba pateks į (0,0052, 0,171) intervalą.

Pažeidimo dažnio kitimas laike



2.9 pav. Apgaubo pažeidimo dažnio kitimo laike funkcijos

Neapibrėžtumo analizė



2.10 pav. Apgaubo pažeidimo dažnio empirinės charakteristikos

Taip pat buvo atlikta pažeidimo dažnio kitimo laike funkcijos neapibrėžtumo analizė. Analizei atlikti sumodeliuotos funkcijos pavaizduotos 2.9 paveiksle. Kaip matome, gautų funkcijų aibė yra gana

plačiai išsibarsčiusi, tačiau paskaičiuota empirinė pažeidimo dažnio mediana (2.21 lentelė) rodo, kad daugiau kaip pusė visų funkcijų maksimali reikšmė neviršija 0,05 ribos. Sudarius itin retą ir nepalankų parametru reikšmių rinkinį, apgaubo pažeidimo tikimybės įvertis nuo $t=7000$ pradeda sparčiai augti ir pasiekia 0,171 ribą. Vertinant sistemos pažeidimo tikimybę su labiausiai tikėtinomis parametru reikšmėmis (2.1 skyrius), gautas rezultatas, 2.10 paveiksle atitinkantis 0,93 empirinį kvantilį, patenka į (0,95, 0,95) tolerancijos intervalą. Apgaubo pažeidimo tikimybės įvertis yra gana toli nuo paskaičiuotos (0,95, 0,95) tolerancijos intervalo viršutinės ribos, todėl realios sistemos pažeidimo tikimybė gali viršyti 0,10787 ribą.

2.1 skyriuje sudaryto modelio rezultatai yra dvigubai didesni už paskaičiuotas vidutines reikšmes atliekant neapibrėžtumo analizę. Tačiau atsižvelgiant į modelio parametru neapibrėžtumą, 2.1 skyriuje nustatytas apgaubo pažeidimo tikimybės įvertis patenka į (0,95, 0,95) tolerancijos intervalą. Kadangi dėl modelio parametru netikslumo pažeidimo dažnis yra išsibarstęs plačiame intervale (0,005; 0,171), būtina tikslinti modelį ir tiksliau įvertinti pažeidimo tikimybę.

2.2.3. MODELIO JAUTRUMO ANALIZĖ

Kad galėtume nustatyti reikšmingiausius modelio parametrus ir pradinis duomenis, atliksime jautrumo analizę. Ji leis identifikuoti tuos parametrus, kurių neapibrėžtumo sumažėjimas leistų patikslinti sudarytą modelį ir tiksliau įvertinti apgaubo pažeidimo tikimybę. Atlikdami jautrumo analizę pirmiausiai paskaičiuosime Pirsono koreliacijos koeficientus (2.17) tarp modelio parametru x_i ir gautų rezultatų y (2.11 paveikslas).

$$R_i = \frac{1/n \sum_j (x_{i,j} - Ex_i)(y_j - Ey)}{\sqrt{\left(n \sum_j x_{i,j}^2 - \left(\sum_j x_{i,j} \right)^2 \right) \left(n \sum_j y_j^2 - \left(\sum_j y_j \right)^2 \right)}}, \quad (2.17)$$

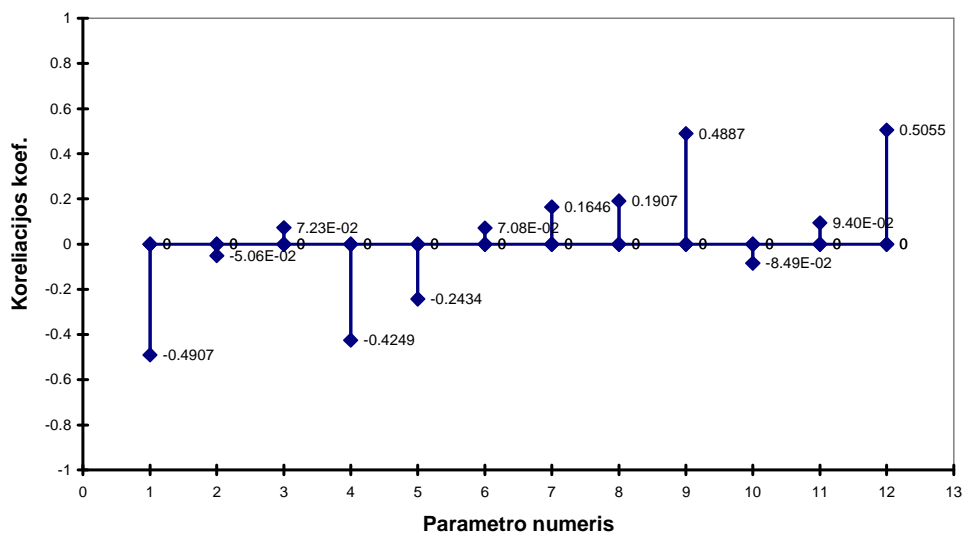
čia $x_{i,j}$ žymi i -ąjį sistemos parametro reikšmę atliekant j -ąjį modeliavimą; Ex_i - x_i parametro vidurkis; Ey - modelio vidurkis; $i = 1, 2, \dots, 12$; $j = 1, 2, \dots, 100$;

Absoliutiniu dydžiu didžiausius koreliacijos koeficientus turi 1, 4, 9 ir 12 parametrai, kurie pagal 2.20 lentelę atitinka pradinį vandens garų kiekį moliais, pirmą pasyviųjų autokatalitinių rekombainerių efektyvumo koeficientą, vandens užliejimo saugos sistemos įjungimo po visiško aktyviosios zonos nusausėjimo tikimybę bei pažeidimo ribos neapibrėžtumo koeficientą. Visų šių parametru koreliacijos koeficientai moduliui yra artimi 0,5, tai rodo vidutinio stiprumo tiesinį ryšį su apgaubo pažeidimo tikimybės įverčiu. Laikoma, kad visi 2.20 lentelėje pateikti modelio parametrai yra tarpusavyje nepriklausomi, tačiau dėl nepakankamai didelės imties gali atsirasti taip vadinama momentinė koreliacija. Ši koreliacija atsiranda tik dėl to, kad esant mažai imčiai neįmanoma užtikrinti,

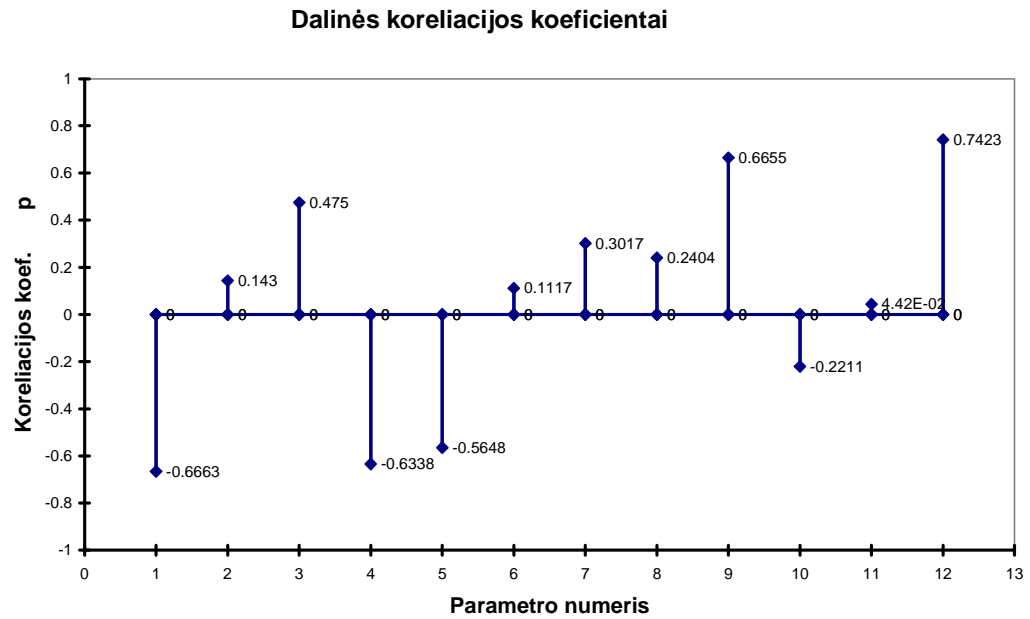
kad ji tenkintų atsitiktinės imties savybes. Momentinė koreliacija gali iškreipti modelio parametru koreliacijos koeficientus, todėl taip pat paskaičiuojami dalinės koreliacijos koeficientai. Dalinės koreliacijos koeficientas atskleidžia rezultato priklausomybę nuo vieno parametro, kai eliminuojama likusių parametru įtaka. Jis yra apibūdinamas kaip koreliacijos koeficientas tarp $y - \hat{y}$ ir $x_i - \hat{x}_i$, $i = 1, 2, \dots, 12$, čia \hat{y} ir \hat{x}_i išreiškiami regresijos lygtimis: $\hat{y} = b_0 + \sum_{h \neq j} b_h x_h$ ir $\hat{x}_i = c_0 + \sum_{h \neq i} c_h x_h$

Paskaičiuotos dalinių koeficientų reikšmės pavaizduotos 2.12 paveiksle. Kaip matome, ankščiau išsiskyre parametrai tik patvirtina savo tiesinę įtaką rezultatams. Vandens garų kiekio dalinis koreliacijos koeficientas įgijo -0,67 reikšmę, taip pat atsiskleidė pasyviųjų autokatalitinių rekombainerių poveikis, tarp apgaubo pažeidimo tikimybės įverčio ir abiejų efektyvumų koeficientų yra vidutinė neigiama tiesinė priklausomybė (-0,63 ir -0,56). Absoliutiniu dydžiu didžiausią dalinės koreliacijos koeficientą turi pažeidimo ribos neapibrėžtumo parametras. Jo reikšmė siekia 0,7423 ir žymi stiprų tiesinį ryšį su modeliavimo rezultatais. Įdomu tai, dalinis koreliacijos koeficientas tarp apgaubo pažeidimo tikimybės ir SIS1 aktyvacijos tikimybės lygus 0,67 ir žymi gana stiprią teigiamą tiesinę priklausomybę, tačiau skaičiavimai rodo, kad išsiskiriančio vandenilio srautas SIS1 aktyvacijos metu, turi labai menką įtaką rezultatams.

Pirsono koreliacijos koeficientai



2.11 pav. Pirsono koreliacijos koeficientai tarp modelio parametru ir pažeidimo dažnio



2.12 pav. Dalinės koreliacijos koeficientai

Apskaičiuoti dalinės koreliacijos koeficientai atskleidžia tiesinius ryšius tarp sistemos parametru ir modelio rezultatų. Tam, kad įsitikintume, jog koreliacijos koeficientai gali būti vertinami, kaip jautrumo įverčiai, papildomai paskaičiuosime standartizuotus regresijos koeficientus (SRK) β_i , $i=1,2,\dots,12$. Standartizuoti regresijos koeficientai atskleidžia, kurie modelio pradiniai duomenys bei parametrai turi didžiausią poveikį apgaubo pažeidimo tikimybei. Jie paskaičiuojami sudarant regresijos funkciją (2.18) tarp normuotų parametru reikšmių $\hat{x}_{i,j}$ ir rezultatų \hat{y}_j .

$$\hat{y}_j = \alpha + \beta_1 \cdot \hat{x}_{j,1} + \dots + \beta_{12} \cdot \hat{x}_{j,12},$$

$$\text{čia } \hat{x}_{i,j} = \frac{x_{i,j} - Ex_i}{\sigma x_i}, \quad \hat{y}_j = \frac{y_j - Ey}{\sigma y} \quad (2.18)$$

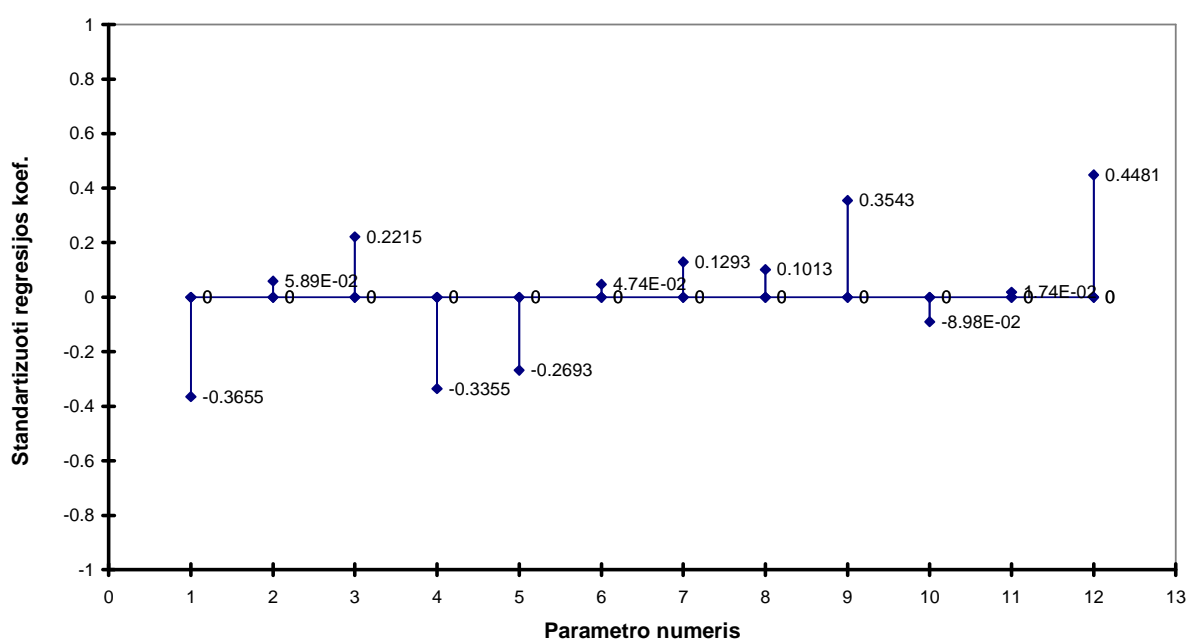
čia $i = 1,2,\dots,12$; $j = 1,2,\dots,100$; Ex_i - x_i parametro vidurkis; Ey - modelio rezultato vidurkis; σx_i - x_i parametro standartinis nuokrypis; σy - modelio rezultato standartinis nuokrypis.

Gautų koeficientų reikšmės pavaizduotos 2.13 paveiksle. Kaip matome, standartizuoti regresijos koeficientai patvirtina dalinių koreliacinių koeficientų rezultatus. Pažeidimo ribos neapibrėžtumo koeficiento SRK turi didžiausią reikšmę 0,4481. Tai rodo, kad pažeidimo ribos neapibrėžtumas turi didžiausią svarbą modelio rezultato neapibrėžtumui. SRK teigiamas ženklas reiškia, kad didesnė pažeidimo ribos neapibrėžtumo koeficiento reikšmė lemia didesnę apgaubo pažeidimo tikimybę. Todėl tobulinant modelį būtina tikslinti apgaubo pažeidimo sąlygas, tiksliau apibrėžti kaip apgaubo pažeidimo tikimybė priklauso nuo sprogo metu susidaręs pikinio slėgio. Kaip rodo paskaičiuoti STR, vandens užliejimo aktyvacijos tikimybės neapibrėžtumas taip pat didina modeliavimo rezultato neapibrėžtumą. Vandens užliejimo tikimybės padidėjimas lemia dažnesnį apgaubo pažeidimą. Jeigu įmanoma valdyti šios saugos sistemos įjungimą, reikia kiek įmanoma sumažinti jos aktyvacijos

tikimybę po visiško aktyviosios zonos nusausėjimo. Iš kitos pusės, didesnis pasyvių katalitinių rekombainerių efektyvumas lemia mažesnę apgaubo pažeidimo tikimybės įvertį. Taip pat skaičiavimai rodo, kad pradinis vandens garų kiekis lemia prastesnes sąlygas sproginams. Šių dviejų parametru neapibrėžtumas taip pat turi įtakos galutinio rezultato netikslumui.

Paskaičiuotas determinacijos koeficientas $R^2=0,84219$ patvirtina, kad SRK turi jautrumo prasmę. Determinacijos koeficientas nurodo, kad 84 procentus sudaryto modelio neapibrėžtumo paaiškina nustatyti parametru neapibrėžtumai. Tai taip pat reiškia, kad sumažinus nustatytų parametru neapibrėžtumus, kartu bus sumažinamas modelio rezultatų neapibrėžtumas.

Standartizuoti regresijos koeficientai



2.13 pav. Standartizuoti regresijos koeficientai

Atlikta jautrumo analizė išskiria penkis reikšmingus modelio parametrus: pradinis vandens garų kiekis, PAR efektyvumo koeficientai, taip pat vandens užliejimo aktyvacijos po visiško aktyviosios zonos nusausėjimo dažnis ir sproginimo neapibrėžtumo koeficientas. Paskutinių dviejų parametru didesnės reikšmės lemia pažeidimo tikimybės padidėjimą, ypač daug dėmesio reiktų skirti apgaubo pažeidimo sąlygos tikslinimui. Jautrumo analizė patvirtina jau patikimumo analizėje padarytą pastebėjimą, kad apgaubo pažeidimo tikimybę padidina SIS1 aktyvacija, todėl valdant šilumnešio praradimo avariją, reiktų vengti jungti šią sistemą po visiško aktyvios zonos nusausėjimo. Taip pat nustatyta, kad apgaubo pažeidimo tikimybę mažina pradinis vandens garų kiekis bei pasyviųjų autokatalitinių rekombainerių efektyvumas.

IŠVADOS IR REZULTATAI

1. Sukurtoje dinaminės sistemos modeliavimo metodikoje atskiriamos determinuotos ir tikimybinės charakteristikos; tikimybinės dinamikos modeliavimo algoritmas suteikia galimybę taikyti metodiką modeliuojant skirtingas sistemas;
2. Sukurta modeliavimo metodika pritaikyta atliekant atominės elektrinės šilumnešio praradimo avarijos ir saugos sistemų patikimumo analizę. Nustatytas apsauginio apgaubo pažeidimo tikimybės įvertis (0,108);
3. Dėl pradinių duomenų ir modelio parametrų neapibrėžtumų, įvertintas sistemos pažeidimo dažnio empirinis maksimumas (0,171) ir minimumas (0,005);
4. Rezultato neapibrėžtumui reikšmingiausias parametras yra apgaubo pažeidimo ribos koeficientas (dalinės koreliacijos koeficientas 0,742), kiti reikšmingi parametrai: vandens užliejimo tikimybė (0,666), pradinis vandens garų kiekis (-0,666), pasyviųjų autokatalitinių rekombainerių efektyvumas (-0,634 ir -0,565);
5. Apsauginio apgaubo pažeidimui lemiamą įtaką turi vandens užliejimo sistemos aktyvacija po visiško aktyviosios zonos nusausėjimo;

PADĖKOS

Noriu padėkoti savo tiriamojo darbo vadovui, dr. Robertui Alzbutui, už įdomią ir aktualią tematiką bei praktiką Lietuvos energetikos institute. Praktikos ir tiriamojo darbo metu įgytos žinios apie saugumo ir rizikos vertinimą, taip pat apie modeliavimo rezultatų analizę, man bus itin naudingos ateityje siekiant tapti geresniu savo srities specialistu. Labai vertinu dr. R. Alzbuto metodinius patarimus ir išsamius komentarus, kurie palengvino baigiamojo darbo rašymo procesą.

Norėčiau padėkoti Fundamentaliųjų mokslų fakulteto dekanui, doc. Vytautui Janiulioniui, už rūpestį bei diskusiją studijų klausimai 2006 metų vasarą. Tik diskusijos metu išgirstų argumentų ir teiginių dėka dabar galiu džiaugtis esamais rezultatais bei savo pasiekimais.

Taip pat tariau nuoširdų ačiū Astai Sirvelytei už išsamų tikimybinės analizės paketo SUSA3.5 aprašymą lietuvių kalba, kurio pagalba sutaupiau nemažai brangaus laiko.

LITERATŪRA

1. Alzbutas R., Kupčiūnienė K., Adlytė R., Augutis J.. Lėktuvo kritimo ant Ignalinos AE tikimybės vertinimas atsižvelgiant į duomenų neapibrėžtumą. Energetika. 2007. Nr. 1. 1-9p.
2. Augutis J. Ušpuras E. Technologijų rizika. Lietuvos energetikos institutas, 2006, 247 p.
3. Ben-Haim, Y. Robust Reliability, Springer Verlag, 1996.
4. Bickel J. H., Kelly D. L. Leahy T. J. Fundamentals of Probabilistic Risk Assessment (PRA). Idaho, 1995.
5. Cornell M. Fault trees and event trees in reliability analysis // Risk Analysis; 1981. Vol 4. No. 3.
6. Čekanavičius V., Murauskas G. Statistika ir jos taikymai. I. Vilnius:TEV, 2000, 240p.
7. Čekanavičius V., Murauskas G. Statistika ir jos taikymai. II. Vilnius:TEV, 2002, 272p.
8. Devooght J. Dynamic reliability // Advances in Nuclear Science and Technology, 1997. Vol. 25, 215-278 p.
9. Dubois, D. and Prade, H. Possibility Theory, Plenum Press, New York, 1988
10. Glaeser H. G., Uncertainty Evaluation of Thermal-Hydraulic Code Results // Proceedings of Int. Meeting on “Best-Estimate” Methods in Nuclear Installation Safety Analysis (BE-2000). Washington DC, USA, 2000
11. Hofer E. Sensitivity analysis in the context of uncertainty analysis for computationally intensive models // Computer Physics Communications. Elsevier Science. 1999. Vol. 117. P. 21–34.
12. Hoyland A., Rausand M. System Reliability Theory. Models and Statistical Methods, John Wiley & Sons, 1994.
13. http://en.wikipedia.org/wiki/Safety_engineering
14. http://en.wikipedia.org/wiki/Probabilistic_risk_assessment
15. Yunwei H. “A guided simulation methodology for dynamic probabilistic risk assessment of complex systems”, 2005, p. 1-13
16. Kloos M., Hofer E., SUSA Version 3.2. User’s Guide and Tutorial. GRS, Garching, 1999.
17. Kopustinskas V., Alzbutas R., Augutis J.. Matematinių modelių rezultatų neapibrėžtumo ir jautrumo parametrams statistiniai tyrimo metodai, Energetika (priimta spaudai), 2007.
18. Kruopis J. Matematinė statistika. Vilnius: Mokslas, 1993, 416.
19. Labeau P.E., Izquierdo J.M., “Modeling PSA Problems-I:The Stimulus Drive Theory of Probabilistic Dynamics”, 2004.
20. Matuzas V., Labeau P.M., “Improvement of event tree methodology using dynamic reliability techniques”, pp. 3-6
21. Modarres M., Kaminskiy M., Krivtsov V. Reliability Engineering and Risk Analysis; A. Practical Guide, Marcel Dekker 1999.

22. Starr C., Rudman R., Whipple C. Philosophical basis for risk analysis. *Ann. Rev. Energy* 1, 629, 1976.
23. Sugeno, M. *Fuzzy Measures and Fuzzy Intervals: A Survey*, 1977.

1 PRIEDAS. PROGRAMINĖ ĮRANGA

```

////////////////////////////////////
// activation
////////////////////////////////////

double Ft (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0:
            {
                // intervale [tmax-5875] SIS sistema ijungiamas su tikimybe 0.5
                if (t < 5875) r = 0.5/(5875-4080)*pas;
                else r = 0;
            } break;
        case 1:
            {
                // intervale [5875-tpstart] SIS sistema ijungiamas su tikimybe 0.5
                if (t >= 5875) r = 0.5/(14220-5875)*pas;
                else r = 0;
            } break;
        case 2:
            {
                if (sys->st->m_h2 >= 950) r = 1;
                else r = 0;
            } break;
        case 3:
            {
                // intervale [tmax-tpstart] CHRS sistema ijungiamas su tikimybe 0.5
                r = 0.5/(tmax-tpstart)*pas;
                if (sys->st->get_pressure() >= 2.4){
                    r=1;
                }
            } break;
        case 4:
            {
                // sproginimo stimulo aktyvavimas
                double h2_conc = sys->st->n_h2 / sys->st->total_moles();
                double h2o_conc = sys->st->n_h2o / sys->st->total_moles();
                double limit1 = 0.04 + 0.04 * h2o_conc + 0.1625 *
                h2o_conc*h2o_conc;
                if (limit1 <= h2_conc){
                    r = 1;
                }
                else r = 0;
            } break;
    }
    return r;
}

////////////////////////////////////
//delay
////////////////////////////////////

double Ht (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0:
            {
                // intervale [tmax-5875] SIS sistema ijungiamas be uzdelsimo,
                // bet kuriuo aktyvacijos laiko momentu
                r = 1;
            }
    }
}

```

```

} break;
case 1:
{
    // intervale [5875-tpstart] SIS sistema ijungiamo be uzdelsimo,
    // bet kuriuo aktyvacijos laiko momentu
    r = 1;
} break;
case 2:
{
    // SIS sistema isjungiamo be uzdelsimo
    // bet kuriuo aktyvacijos laiko momentu
    r = 1;
} break;
case 3:
{
    // intervale [tmax-tpstart] CHRS sistema ijungiamo be uzdelsimo,
    // bet kuriuo aktyvacijos laiko momentu
    r = 1;
} break;
case 4:
{
    // sprogimo stimulo aktyvavimas
    double h2_conc = sys->st->n_h2 / sys->st->total_moles();
    double h2o_conc = sys->st->n_h2o / sys->st->total_moles();
    double limit1 = 0.04 + 0.0402 * h2o_conc + 0.1625 *
h2o_conc*h2o_conc; //??
    double limit2 = 0.052 + 0.0964 * h2o_conc;
    if ((h2_conc > 0.9999*limit1) && (h2_conc < 1.0001*limit1))
        r = pas/14400;
    else if ((h2_conc < limit2) && (h2_conc > limit1))
    {
        double dt;
        dt = (14400*(h2_conc-limit2)-1200*(h2_conc-
limit1))/(limit1-limit2);
        r = pas/dt;
    }else if (h2_conc > limit2)
        r = pas/1200;
    else r = 0;
} break;
}
return r;
}

////////////////////////////////////
//deactivation
////////////////////////////////////

double Kt (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0: {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 1:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 2:
        {
            r = 0;
        } break;
        case 3:

```

```

    {
        // negalima isjungti sistemas
        r = 0;
    } break;
case 4:
    {
        // sprogimo stimulo deaktyvavimas
        double h2_conc = sys->st->n_h2 / sys->st->total_moles();
        double h2o_conc = sys->st->n_h2o / sys->st->total_moles();
        double limit1 = 0.04 + 0.04 * h2o_conc + 0.0016 *
h2o_conc*h2o_conc;
        if (h2_conc < limit1)
            r = 1;
        else
            r = 0;
    } break;
}
return r;
}

////////////////////////////////////
//delay deactivation
////////////////////////////////////

double Lt (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 1:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 2:
        {
            r = 0;
        } break;
        case 3:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 4:
        {
            // sprogimo stimulo deaktyvavimas be uzdelsimo
            r = 1;
        } break;
    }
    return r;
}

#include <cstdlib>
#include "StimulSim.h"

////////////////////////////////////
// Code for simulation based on stimulus driven teory of probabilistic dynamics
////////////////////////////////////

main()
{

```



```

// long _s, _f , nbrupt;
//      read_params(); // inputs params from file

N = 10000;
combs = new MyList();
time(&_s); // moment of simulation start

stisimrun();

time(&_f); // moment of simulation finish

cout << "The program starts at: " << _s % 86400 / 3600 +1 << " h ";
cout << _s % 3600 / 60 << " min " << _s % 60 << " sec" <<endl<<endl;

cout << "The program finishes at: " << _f % 86400 / 3600 +1 << " h ";
cout << _f % 3600 / 60 << " min " << _f % 60 << " sec" <<endl<<endl;

stisimout();

cout << "----- " << endl;
cout << "Rezultatai stisimout.csv faile. " << endl;
cin >> nbrupt;

delete combs;

//      sampling(); // sampling for S&U analysis

}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// FAILURES SIMULATION BASED ON STIMULUS DRIVEN DYNAMICS
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

void initiation()
{
    /* initial conditions */
    tpstart=4080;

    tpstrans=tpstart;
    for(int sa=0; sa<nbstim; sa++)
    {
        etact[sa]=0;
        etdes[sa]=0;
        tpsact[sa]=0;
        tpsdes[sa]=0;

        a[sa]=0;
        da[sa]=0;
        d[sa]=0;
        dd[sa]=0;

        etdelact[sa]=0;
        etdeldes[sa]=0;
    }

    // trinti
    for(int b=0; b<nbvar; b++)
    {
        xtrans[b]=0;
        zero[b]=0;
    }
}

```

```

    }
//end: trinti
    for(int b1=0; b1<nbstim; b1++)
    {
        State *new_st = new State();
        new_st->get_instance(xact[b1]);

        delete(new_st);

    }
//end: trinti????

/* initialization */

    newdyn=dyn;

    stop=0;
    nons=0;
    nonsd=0;

/* determination of the "probabilities thresholds"
    act[0] = 0.687;
    act[1] = 0.574;
    act[2] = 0.229;
    act[3] = 0.584;
    act[4] = 0.278;

    delact[0] = 0.015;
    delact[1] = 0.015;
    delact[2] = 0.473;
    delact[3] = 0.345;
    delact[4] = 0.836;

    desact[0] = 0.825;
    desact[1] = 0.825;
    desact[2] = 0.729;
    desact[3] = 0.285;
    desact[4] = 0.175;

    deldes[0] = 0.157;
    deldes[1] = 0.157;
    deldes[2] = 0.078;
    deldes[3] = 0.503;
    deldes[4] = 0.788;          */

    for (int st=0; st<nbstim; st++)
    {
        act[st]=daleatory(0,1);
        delact[st]=daleatory(0,1);
        desact[st]=daleatory(0,1);
        deldes[st]=daleatory(0,1);
    }
}
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// FAILURES SIMULATION BASED ON STIMULUS DRIVEN DYNAMICS
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

void stisimrun()
{
    srand(time(NULL)); /* essential for the generation of random numbers */

    /* initialization of the vector of the results */

```

```

for(int p=0; p<proba_size;p++){
    proba[p]=0;
}

for(int i=0; i<scenar_num; i++){
    scenarios[i]=0;
}

// creating dynamic system
System *sys = new System();
sys->write_log = 0;

/* For each history... */
int c = 0;
for (int i=0; i<N; i++)
{

cout << " history: " << i << endl;
sys->init();
initiation();

/* For each time step ... */
for (double t=tpstart+pas; t<(tmax +(pas/2)) && stop==0; t=t+pas)
{
    /* calculation of the physical variables */
    sys->evol(t);
    // cout << t;
    /* For each stimulus in defined order... */

    orderstim();

    for (int w=0; w<nbstim && stop==0; w++)
    {
        int s = stimorder[w]; // the considered stimulus

        /* If the stimulus is not activated */
        if(etact[s]==0)
        {
            /* activation initiation probability*/
            a[s]=a[s]+Ft(sys,s,t);

            /* if this probability is higher than */
            /* the threshold: initiate activation */
            if(a[s]>=act[s])
            {
                /* initiate activation */
                etact[s]=1;

                /* memorize variables */
                tpsact[s]=t;
                sys->st->get_instance(xact[s]);
                sys->active_stimul(s);

                // deinitiate deactivation
                etdes[s]=0;
                d[s]=0;
                dd[s]=0;
                etdeldes[s]=0;
            }
        }

        /* If the stimulus is activated */

```

```

if(etact[s]==1)
{
    // choice for "deactivation first" or
    // "end of the activation delay first"
    int z=aleatory(0,1);

    /* If no domination of deactivation */
    if(z==0)
    {
        /* %%% end of delay %%% */
        ifdelact0(sys, s,t);

        /* %%% deactivation %%% */
        ifdes0(sys,s,t);
        if (etdeldes[s]==0)
            ifdes1(sys,s,t);
    }

    /* Deactivation dominates over delay */
    if(z==1)
    {
        /* %%% deactivation %%% */
        ifdes0(sys,s,t);
        if (etdeldes[s]==0)
            ifdes1(sys,s,t);

        /* %%% end of delay %%% */
        if (etact[s]==1)
            ifdelact0(sys,s,t);
    }
}
int status = sys->get_status();

if(status > 0){
    stop=1;
    c++;
}
if(status < 0)
{
    nbrupt=nbrupt+1;
    stop=1;

    int k=(t-proba_start-1)/proba_step;
    for(; k<proba_size ; k++)
    {
        proba[k]=proba[k]+1;
    }
}
}

combs->add(sys->combs);
scenarios_comb[sys->scenario].add(sys->combs);
scenarios_dynam[sys->scenario].add(sys->dynam, (sys->get_status()<0));
scenarios[sys->scenario]++;
}
cout << "pazeistas tiek kartu: " << nbrupt << endl;
cout << "sprogimas ivyko: " << combs->count << endl;

//      for (int k = 0; k<scenar_numb; k++){
//          cout << k << ": " << 1.0*scenarios_comb[k].count << " : " <<
1.0*scenarios_comb[k].average() <<endl;
//          cout << k << ": " << scenarios_comb[k].rupt_count <<"/" <<scenarios[k] <<
"/" << scenarios_comb[k].count << endl;

```



```

void ifdelact0(System *sys, int s, double t)
{
    if(etdelact[s]==0)
    {
        /* calculation of probability for end of delay */
        da[s]=da[s]+Ht(sys,s,t);

        /* if this probability > threshold */
        if(da[s]>=delact[s])
        {
            /* activation delay finishing */
            etdelact[s]=1;

            /* choice of the new dynamics */
            newdynamics(sys, s,t);
        }
    }
}

////////////////////////////////////
// Check (stimulus of) deactivation before initiation
////////////////////////////////////

void ifdes0(System *sys, int s, double t)
{
    if(etdes[s]==0)
    {
        //cout << "Deactivation initiation.. " << endl;

        /* calculation of deactivation initiation probability */
        d[s]=d[s]+Kt(sys,s,t);

        /* if this probability > threshold */
        if(d[s]>=desact[s])
        {
            /* deactivation initiation */
            etdes[s]=1;

            /* memorize variables
            if(nonsd==0)
            {
                tpsdes[s]=t;
                for(int i=0; i<nbvar; i++)
                {
                    xdes[s][i]=x[i];
                }
            }
            else nonsd=0;
            */
        }
    }
}

////////////////////////////////////
// Check (stimulus of) deactivation after it initiation
////////////////////////////////////

void ifdes1(System *sys, int s, double t)
{
    if(etdes[s]==1)
    {
        /* calculation of deactivation delay time*/
        dd[s]=dd[s]+Lt(sys,s,t);
    }
}

```

```

/* if this probability is higher than the threshold */
if(dd[s]>=deldes[s])
{
    /* deactivation delay finishing */
    etdeldes[s]=1;

    /* activation stoping */
    etact[s]=0;
    etdelact[s]=0;

    /* memorize variables */
    tpsact[s]=t;
    sys->st->get_instance(xact[s]);

    /* clear probabilities */
    a[s]=0;
    da[s]=0;
    d[s]=0;
    dd[s]=0;

    /* determination of new "probabilities thresholds" */
    act[s]=daleatory(0,1);
    delact[s]=daleatory(0,1);
    desact[s]=daleatory(0,1);
    deldes[s]=daleatory(0,1);
}
}
}

////////////////////////////////////
// Output
////////////////////////////////////

void stisimout()
{
    char *scenarios_name = "ABCDE";
    /* opening of the files for writing */

    ofstream output;
    output.open("stisimout.csv"); /* outputs */

    /* writing the hour of beginning in the file of results */
    output << "Generacijų kiekis: ;" << N << ';' << endl;
    output << "Pažeidimu kiekis: ;" << nbrupt << ';' << endl;

    output << ";;" << endl;
    output << ";;" << endl;
    output << ";;" << endl;
    output << "Pažeidimų tikimybės;" << endl;
    output << "Laikas; Pažeidimo tikimybė;" << endl;

    double t = proba_start+proba_step;
    for (int k = 0; k<proba_size; k++){
        output << t << ";"<< 1.0*proba[k]/N << "; " << endl;
        t +=proba_step;
    }

    output << ";;" << endl;
    output << ";;" << endl;
    output << ";;" << endl;

    output << "Scenarijų analizė;" << endl;

```

```

output << "Scenarijus; Pasirodymo tikimybė; " << endl;

for (int k = 0; k<scenar_num; k++){
    output << scenarios_name[k] << " "; " << 1.0*scenarios[k]/N << " "; "<<
endl;
}

output << ";;" << endl;
output << ";;" << endl;
output << ";;" << endl;

output << "Scenarijus; Pažeidimų kiekis; Sprogimų kiekis; Pasirodymų
kiekis;" ;
output << "Sprogimo stiprumo vidurkis; Sprogimo stiprumo dispersija;"<<
endl;

for (int k = 0; k<scenar_num; k++){
    output << scenarios_name[k] << " "; " << scenarios_comb[k].rupt_count
<< " "; " ";
    output << scenarios_comb[k].count << " "; " << scenarios[k] << " "; " ";
    output << scenarios_comb[k].average() << " "; " <<
scenarios_comb[k].variance() << " "; " << endl;
    cout << scenarios_comb[k].average() << " "; " <<
scenarios_comb[k].variance() << " "; " << endl;
}

output << ";;" << endl;
output << ";;" << endl;
output << ";;" << endl;

output << "Dinamikų perėjimų grupavimas;" << endl;
output << "Scenarijus; Dinamikų sekos; Pasirodymų kiekis; Pažeidimų
kiekis;" << endl;
for (int k = 0; k<scenar_num; k++){
    DynamList *start = scenarios_dynam[k].first;
    while (start !=NULL){
        output << scenarios_name[k] << " "; " << start->dynamic << " ";
";
        output << start->history_count << " "; " << start-
>rupture_count << " "; " << endl;
        start = start->next;
    }
}

output.close();
}

////////////////////////////////////
// SAMPLING FOR U&S ANALYSIS
////////////////////////////////////

void sampling()
{
    // Sampling data for uncertain parameters are in file: stisimin.sam
    // Modeling results using sampling data are in faile: stisimout.srs

//    cout << endl << "SIMULATION" << endl;

/* essential for the generation of random numbers */

//    srand(time(NULL));

```



```

        /* opening of the files of reading and writing */

//      ifstream _2input;
//      _2input.open("stisimin.sam"); /* sampling parameters */
//      ofstream _2output;
//      _2output.open("stisimout.srs"); /* sampling results */

//      cout << "Sampling parameters in file: stisimin.sam" << endl;
//      cout << "Sampling results in file: stisimout.srs" << endl;

////////// RUNING

        /* writing the hour of beginning in the file of results */

//      time(&_s);
//      cout << "The program starts at: " << _s % 86400 / 3600 + 1 << " h ";
//      cout << _s % 3600 / 60 << " min " << _s % 60 << " sec" <<endl<<endl;

//      while (!_2input.eof()){

                /* input values of model parameters */
                //      _2input >> pvmin; _2input >> kv;

                //      cout << "Parameter pvmin: " << pvmin << endl;
                //      cout << "Parameter kv: " << kv << endl;

                //pressakey();
                /* clear vector of results story */
//      for(int i=0; i<proba_size;i++)
//      {
//      proba[i]=0;
//      }
                nbrupt=0;
                ///

                stisimrun();

                ///
//      _2output << tailleproba << endl;
//      float t = tpstart;
//      _2output << t << ' ' << proba[0] << endl;
//      for(int i=1; i<tailleproba;i++)
//      {
//      proba[i]=proba[i]+proba[i-1];
//      t = t + pas;
//      _2output << t << ' ' << proba[i] << endl;

//      }
//      }

//      time(&_f);
//      cout << "The program ends at: " << _f % 86400 / 3600 + 1 << " h ";
//      cout << _f % 3600 / 60 << " min " << _f % 60 << " sec" <<endl<<endl;

//      _2output.close();

}

class State{

```

```

public:
    double n_h2;
    double n_o2;
    double n_h2o;
    double n_n2;
    double m_h2;
    double temp;
    double p;
    double t;
public:
    State();
    ~State();
    void initiate();
    void get_instance(State &st);
    double State::get_pressure();
    double total_moles();
};
// constructor
State::State(){
    this->initiate();
}
// destructor
State::~State(){}

// initiate parameters
void State::initiate(){
    n_h2 = 0.0;
    n_o2 = 368384;
    n_h2o = 1717196;
    n_n2 = 1390372;
    m_h2 = 0;
    temp = 382.05;
    p = 2.34;
    t = 4080;
}
void State::get_instance(State &st){
    st.n_h2 = this->n_h2;
    st.n_h2o = this->n_h2o;
    st.n_o2 = this->n_o2;
    st.n_n2 = this->n_n2;
    st.m_h2 = this->m_h2;
    st.temp = this->temp;
    st.p = this->p;
    st.t = this->t;
}
double State::total_moles(){
    return this->n_h2 + this->n_h2o + this->n_o2 + this->n_n2;
}
double State::get_pressure(){
    this->p=2.34*total_moles()/3475952;
    return this->p;
}

class System{
public:
    int sis_on;
    int chrs_on;
    State* st;
    double peak_pressure;

    int write_log;
    int scenario;
    MyList *combs;
    string dynam;

```

```

    System();
    ~System();
    void System::evol(double t);
    void System::init();
    int System::get_status();
    void System::active_stimul (int i);
    void System::active_event (int i);
private:
    double get_h2(double t, State *s);
    double get_temp(double t);
    double get_delta_h2(double t, State *s);
    double get_delta_h2_Zr_oxid(double t, State* s);
    double get_delta_o2(double t, State *s);
    double get_delta_h2o(double t, State *s);

    void combustion();
};
// constructor
System::System(){
    st = new State();
    combs = NULL;
    this->init();
    write_log = 0;
}
// destructor
System::~~System(){
    delete st;
    if (combs != NULL) delete combs;
}

void System::init(){
    st->initiate();
    sis_on = -1;
    chrs_on = -1;
    peak_pressure = 0;

    scenario = 0;
    dynam = "0 ";
    if (combs != NULL) delete combs;
    combs = new MyList();

    if (write_log > 0){
        ofstream p_log, h2_log, event_log;
        p_log.open("pressure.out");
        h2_log.open("h2_con.out");
        event_log.open("event.out");
        p_log.close();
        h2_log.close();
        event_log.close();
    }
}

void System::evol(double t){
    // counting the hydrogen mass in the containment without water injection
    // Assumption 2
    double t_old = st->t;
    double delta = t - t_old;
    double dt = delta/10;
    double delta_h2, delta_o2, delta_h2o, delta_h2_Zr;

```

```

double delta_h2_old = this->get_delta_h2(t_old, st);
double delta_o2_old = this->get_delta_o2(t_old, st);
double delta_h2o_old = this->get_delta_h2o(t_old, st);
double delta_h2_Zr_old = this->get_delta_h2_Zr_oxid(t_old, st);
double n_h2, n_o2, n_h2o, m_h2;

// nepamirsti apie vienetu suderinamuma

for (double tt=t_old+dt; tt<t+dt/2;tt=tt+dt){
    delta_h2 = this->get_delta_h2(tt, st);
    n_h2 = st->n_h2 + (delta_h2_old+delta_h2)*0.5*dt;
    delta_h2_old = delta_h2;

    delta_o2 = this->get_delta_o2(tt, st);
    n_o2 = st->n_o2 + (delta_o2_old+delta_o2)*0.5*dt;
    delta_o2_old = delta_o2;

    delta_h2o = this->get_delta_h2o(tt, st);
    n_h2o = st->n_h2o + (delta_h2o_old+delta_h2o)*0.5*dt;
    delta_h2o_old = delta_h2o;

    delta_h2_Zr = this->get_delta_h2_Zr_oxid(tt, st);
    m_h2 = st->m_h2 + (delta_h2_Zr+delta_h2_Zr_old)*0.5*dt;
    delta_h2_Zr_old = delta_h2_Zr;

    st->n_h2 = n_h2;
    st->n_o2 = n_o2;
    st->n_h2o = n_h2o;
    st->m_h2 = m_h2;
    st->temp = this->get_temp(tt);
}
// cout << st->n_h2 << '-' << st->total_moles() << endl;
st->t = t;

if (write_log > 0){
    ofstream h2_log, p_log;
    h2_log.open("h2_con.out", 1);
    h2_log << st->n_h2/st->total_moles() << endl;
    h2_log.close();

    p_log.open("pressure.out", 1);
    p_log << st->get_pressure() << endl;
    p_log.close();
}
}
double System::get_delta_h2_Zr_oxid(double t, State* s){
    double delta = 0;

    if (t <= 4081) delta = 0;
    else if (t <= 5340) delta = 0.00066*(t-4081)/(5340-4081);
    else if (t <= 5660) delta = 0.00066 + (0.028-0.00066)*(t-5340)/(5660-
5340);
    else if (t <= 5970) delta = 0.028 + (0.17-0.028)*(t-5660)/(5970-5660);
    else if (t <= 6110) delta = 0.17 + (0.36-0.17)*(t-5970)/(6110-5970);
    else if (t <= 6425) delta = 0.36 + (0.156-0.36)*(t-6110)/(6425-6110);
    else if (t <= 7005) delta = 0.156 + (0.04-0.156)*(t-6425)/(7005-6425);
    else if (t <= 7410) delta = 0.04 + (0.09-0.04)*(t-7005)/(7410-7005);
    else if (t <= 9180) delta = 0.09 + (0.034-0.09)*(t-7410)/(9180-7410);
    else if (t <= 11070) delta = 0.034 - 0.034*(t-11070)/(11070-9180);
    else delta = 0;

    // adding mass produced by steel and Zr oxidation
    if ((this->sis_on != -1) && (this->sis_on <= t) && (t > 5875)){

```

```

        delta += 0.833;
    }
    return delta;
}

double System::get_delta_h2(double t, State* s){
    double delta = this->get_delta_h2_Zr_oxid(t, s);

    // subtracting recombined
    delta -= (3*s->p+3.7)*s->n_h2/s->total_moles()*0.1; // 100*0.0001
    (conversion to kg)

    return delta/2.01594*1000;
}

double System::get_delta_o2(double t, State *s){
    double delta = 0;

    delta -= (3*s->p+3.7)*s->n_h2/s->total_moles()*800; // 8*100

    return delta/31.9988;
}

double System::get_delta_h2o(double t, State *s){
    double delta = 0;

    if ((this->chrs_on != -1) && (this->chrs_on <= t)){
        double tau = t - this->chrs_on;

        //if (s->n_h2o > 1717196) tau = 0;
        //else if (s->n_h2o > 968499) tau = 0 + 600*(1717196-s->n_h2o)/(1717196-968499);
        //else if (s->n_h2o > 416979) tau = 600 + (1500-600)*(968499-s->n_h2o)/(968499-416979);
        //else if (s->n_h2o > 232292) tau = 1500 + (2700-1500)*(416979-s->n_h2o)/(416979-232292);
        //else if (s->n_h2o > 201399) tau = 2700 + (8000-2700)*(232292-s->n_h2o)/(232292-201399);
        //else tau = 8000;

        if (tau <= 600) delta += (968499.0-1717196.0)/600.0;
        else if (tau <= 1500) delta += (416979.0-968499.0)/(1500.0-600.0);
        else if (tau <= 2700) delta += (232292.0-416979.0)/(2700.0-1500.0);
        else delta += (201399.0-232292.0)/(8000.0-2700.0);
    }
    return delta;
}

double System::get_temp(double t){
    double temp = 108.9;
    if ((this->chrs_on == -1) || (this->chrs_on > t)) return temp+273.15;

    // counting the average temperature of gas in the containment after the
    chrs activation
    // Assumpltion 7.
    double delta = t - this->chrs_on;
    if (delta <= 600) temp = 108.9 + (91-108.9)*delta/600;
    else if (delta <= 1500) temp = 91 + (73-91)*(delta-600)/(1500-600);
    else if (delta <= 2700) temp = 73 + (61-73)*(delta-1500)/(2700-1500);
}

```

```

else if (delta <= 8000) temp = 61 + (55-61)*(delta-2700)/(8000-2700);
else if (delta <= 13000) temp = 55 + (50-55)*(delta-8000)/(13000-8000);
else temp = 55;

return temp+273.15;
}

void System::combustion(){
double comb_fraction = daleatory(0.05,1);
if (st->n_o2 < 0.5*comb_fraction*st->n_h2){
comb_fraction = 2.0*st->n_o2/st->n_h2;
}
double total = st->total_moles();

this->peak_pressure = aicc(st->temp, st->get_pressure(), st->n_h2/total*100,
st->n_h2o/total*100, st->n_o2/total*100, st->n_n2/total*100, comb_fraction);
st->n_o2 = st->n_o2 - 0.5*comb_fraction*st->n_h2;
st->n_h2o = st->n_h2o + comb_fraction*st->n_h2;
st->n_h2 = st->n_h2 - comb_fraction*st->n_h2;

combs->add(this->peak_pressure);
if (write_log > 0){
ofstream p_log;
p_log.open("pressure.out", 1);
p_log << this->peak_pressure << endl;
p_log.close();
}
}

int System::get_status(){
if ((sis_on != -1)&&(sis_on < 5875)) return 1;
int status = 0;
//if (this->peak_pressure > 5) status = -1;
//else status = 0;

double failure = daleatory (0, 1);
double failure_limit = 1;
//int status = 0;

if (this->peak_pressure < 3) failure_limit = 0;
else if (this->peak_pressure < 4) failure_limit = 0 + 0.02*(this->
peak_pressure-3);
else if (this->peak_pressure < 5) failure_limit = 0.02 + (0.05-
0.02)*(this->peak_pressure-4);
else if (this->peak_pressure < 6) failure_limit = 0.05 + (0.10-
0.05)*(this->peak_pressure-5);
else if (this->peak_pressure < 7) failure_limit = 0.10 + (0.50-
0.10)*(this->peak_pressure-6);
else if (this->peak_pressure < 8) failure_limit = 0.50 + (0.75-
0.50)*(this->peak_pressure-7);
else if (this->peak_pressure < 9) failure_limit = 0.75 + (0.90-
0.75)*(this->peak_pressure-8);
else if (this->peak_pressure < 10) failure_limit = 0.90 + (0.1-
0.90)*(this->peak_pressure-9);
else failure_limit = 1;

if (failure <= failure_limit) status = -1;
else status = 0;

this->peak_pressure = 0;

return status;
}

void System::active_stimul (int i){

```

```

    if (write_log > 0){
        ofstream event_log;
        event_log.open("event.out", 1);
        event_log << st->t - tpstart <<" : Stimul " << i << endl;
        event_log.close();
    }

}

void System::active_event (int i){
    if (write_log > 0){
        ofstream event_log;
        event_log.open("event.out", 1);
        event_log << st->t - tpstart << " : event " << i << endl;
        event_log.close();
    }

    switch(i){
        case 0: {
            sis_on = st->t;
            if (scenario == 0) scenario = 4;
            if (scenario == 1) scenario = 2;
            dynam += "1 ";
        } break;
        case 1: {
            sis_on = st->t;
            if (scenario == 0) scenario = 4;
            if (scenario == 1) scenario = 2;
            if (chrs_on == -1) dynam += "2 ";
            else dynam += "4 ";
        } break;
        case 2: {
            sis_on = -1;
            if (chrs_on == -1) dynam += "0 ";
            else dynam += "3 ";
        } break;
        case 3: {
            chrs_on = st->t;
            if (scenario == 0) scenario = 1;
            if (scenario == 4) scenario = 3;
            if (sis_on == -1) dynam += "3 ";
            else dynam += "4 ";
        } break;
        case 4:{
            this->combustion();
            dynam += "5 ";
            if ((chrs_on == -1) && (sis_on == -1)) dynam += "0 ";
            if ((chrs_on == -1) && (sis_on != -1)) dynam += "2 ";
            if ((chrs_on != -1) && (sis_on == -1)) dynam += "3 ";
            if ((chrs_on != -1) && (sis_on != -1)) dynam += "4 ";
        }
    }
}

}

// activation
// activation

```

```

double Ft (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0:
            {
                // intervale [tmax-5875] SIS sistema ijungiamas su tikimybe 0.5
                if (t < 5875) r = 0.5/(5875-4080)*pas;
                else r = 0;
            } break;
        case 1:
            {
                // intervale [5875-tpstart] SIS sistema ijungiamas su tikimybe 0.5
                if (t >= 5875) r = 0.5/(14220-5875)*pas;
                else r = 0;
            } break;
        case 2:
            {
                if (sys->st->m_h2 >= 950) r = 1;
                else r = 0;
            } break;
        case 3:
            {
                // intervale [tmax-tpstart] CHRS sistema ijungiamas su tikimybe 0.5
                r = 0.5/(tmax-tpstart)*pas;
                if (sys->st->get_pressure() >= 2.4){
                    r=1;
                }
            } break;
        case 4:
            {
                // sprogimo stimulo aktyvavimas
                double h2_conc = sys->st->n_h2 / sys->st->total_moles();
                double h2o_conc = sys->st->n_h2o / sys->st->total_moles();
                double limit1 = 0.04 + 0.04 * h2o_conc + 0.1625 *
                h2o_conc*h2o_conc;
                if (limit1 <= h2_conc){
                    r = 1;
                }
                else r = 0;
            } break;
    }
    return r;
}

//delay

double Ht (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0:
            {
                // intervale [tmax-5875] SIS sistema ijungiamas be uzdelsimo,
                // bet kuriuo aktyvacijos laiko momentu
                r = 1;
            } break;
        case 1:
            {
                // intervale [5875-tpstart] SIS sistema ijungiamas be uzdelsimo,
                // bet kuriuo aktyvacijos laiko momentu
                r = 1;
            } break;
    }
}

```



```

case 2:
{
    // SIS sistema isjungiamo be uzdelsimo
    // bet kuriuo aktyvacijos laiko momentu
    r = 1;
} break;
case 3:
{
    // intervale [tmax-tpstart] CHRS sistema iungiamo be uzdelsimo,
    // bet kuriuo aktyvacijos laiko momentu
    r = 1;

} break;
case 4:
{
    // sprogimo stimulo aktyvavimas
    double h2_conc = sys->st->n_h2 / sys->st->total_moles();
    double h2o_conc = sys->st->n_h2o / sys->st->total_moles();
    double limit1 = 0.04 + 0.0402 * h2o_conc + 0.1625 *
h2o_conc*h2o_conc; //??
    double limit2 = 0.052 + 0.0964 * h2o_conc;
    if ((h2_conc > 0.9999*limit1) && (h2_conc < 1.0001*limit1))
        r = pas/14400;
    else if ((h2_conc < limit2) && (h2_conc > limit1))
    {
        double dt;
        dt = (14400*(h2_conc-limit2)-1200*(h2_conc-
limit1))/(limit1-limit2);
        r = pas/dt;
    } else if (h2_conc > limit2)
        r = pas/1200;
    else r = 0;
} break;
}
return r;
}

//////////////////////////////////////
//deactivation
//////////////////////////////////////

double Kt (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0: {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 1:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 2:
        {
            r = 0;

        } break;
        case 3:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 4:
        {
            // sprogimo stimulo deaktyvavimas
            double h2_conc = sys->st->n_h2 / sys->st->total_moles();

```

```

        double h2o_conc = sys->st->n_h2o / sys->st->total_moles();
        double limit1 = 0.04 + 0.04 * h2o_conc + 0.0016 *
h2o_conc*h2o_conc;
        if (h2o_conc < limit1)
            r = 1;
        else    r = 0;
    } break;

}
return r;

}

////////////////////////////////////
//delay deactivation
////////////////////////////////////

double Lt (System *sys, int stimul, double t)
{
    double r = 0;
    switch(stimul){
        case 0:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 1:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 2:
        {
            r = 0;

        } break;
        case 3:
        {
            // negalima isjungti sistemas
            r = 0;
        } break;
        case 4:
        {
            // sprogimo stimulo deaktyvavimas be uzdelsimo
            r = 1;
        } break;
    }
    return r;
}

```