



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS**

NERIJUS BALTAITIS

**MASĖS PERDAVIMO PROCESŲ BIOREAKTORIŲ
SISTEMOSE TYRIMAS**

Baigiamasis magistro projektas

Vadovas

Prof. Vytautas Galvanauskas

KAUNAS, 2015

**KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS
AUTOMATIKOS KATEDRA**

**MASĖS PERDAVIMO PROCESŲ BIOREAKTORIŲ
SISTEMOSE TYRIMAS**

Baigiamasis magistro projektas
Valdymo technologijos (kodas 621H66001)

Vadovas

Prof. Vytautas Galvanauskas

Recenzentas

Projektą atliko

Nerijus Baltaitis

KAUNAS, 2015



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Elektros ir elektronikos

(Fakultetas)

Nerijus Baltaitis

(Studento vardas, pavardė)

Valdymo technologijos, 621H66001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Masės perdavimo procesų bioreaktorių sistemose tyrimas“

AKADEMINIO SAŽININGUMO DEKLARACIJA

20 15 m. gegužės 26 d.
Kaunas

Patvirtinu, kad mano **Nerijaus Baltaičio** baigiamasis projektas tema „Masės perdavimo procesų bioreaktorių sistemose tyrimas“ yra parašytas visiškai savarankiškai, o visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

Baltaitis, N. Masės perdavimo procesų bioreaktorių sistemose tyrimas. Magistro baigiamasis projektas / vadovas prof. Vytautas Galvanauskas; Kauno technologijos universitetas, elektros ir elektronikos fakultetas, automatikos katedra.

Kaunas, 2015. 70 psl.

SANTRAUKA

Šiame baigiamajame magistro projekte tyrinėjamas masės perdavimo procesas bioreaktorių sistemose. Naudojant dviejų realių procesų duomenis, stacionarios būsenos metodu apskaičiuotos masės perdavimo iš dujinės į skystąją fazę koeficiento $k_L a$ reikšmės. Sukurti dirbtinių neuronų tinklų modeliai Matlab programavimo aplinkoje, naudojant NNSYSID ir NNTOOL paketus. Pasitelkus juodosios dėžės modelius, identifikuotos skirtingų procesų masės perdavimo $k_L a$ reikšmės. Atliktas dirbtinių neuronų tinklų prognozavimo kokybės palyginimas. Išanalizuota neuronų skaičiaus paslėptame sluoksnyje įtaka prognozavimo kokybei. Gauti rezultatai rodo, kad dirbtinių neuronų tinklas gali būti taikomas $k_L a$ reikšmės įvertinimui skirtingoms fermentacijos sistemoms. Sukurti modeliai toliau gali būti naudojami atliekant modelių pagrįstą bioproceso optimizavimą, kai masės perdavimo procesų intensyvumas yra vienas iš proceso našumą ribojančių veiksnių.

Reikšminiai žodžiai : maišymas, aeracija, dirbtinis neuronų tinklas, bioreaktorius, daugiasluoksnis perceptronas, juodosios dėžės modelis, deguonies masės perdavimas, stacionarios būsenos metodas

Baltaitis, N. Investigation of Mass Transfer Processes in Bioreactor Systems. Final project of master degree / supervisor Prof. Vytautas Galvanauskas; Kaunas University of Technology, Faculty of Electrical and Electronics Engineering, Department of Automation.

Kaunas, 2015. 70 p.

SUMMARY

In this final master's project the mass transfer process in bioreactor systems is investigated. Using data stationary state method, for two real processes the values of the coefficient of mass transfer from gas to liquid phase ($k_L a$) are calculated. Artificial neural network models were developed in Matlab programming environment using NNSYSID and NNTOOL packages. With the help of black box models, the values of $k_L a$ in a different mass transfer processes were

identified. The comparison of artificial neural network forecasting quality was performed. The impact of the number of hidden layer neurons on the quality of forecasting was analyzed. The results show that the artificial neural network can be applied for the k_La value assessment in the different fermentation systems. The created models can be used for model-based optimization of bioprocesses when an intensity of mass transfer processes is one of the factors limiting the performance of the process.

Keywords: mixing, aeration, artificial neural network, bioreactor, Multilayer Perceptron, black box model, mass transfer of oxygen, the stationary state method

Turinys

SANTRUMPŲ IR ŽENKLŲ AIŠKINIMO ŽODYNAS	7
ĮVADAS	8
1. APŽVALGINĖ DALIS	10
1.1 Bioreaktoriai	10
1.1.1 Aerliftiniai bioreaktoriai	11
1.1.2 Bioreaktoriai su maišyklėmis	13
1.1.3 Burbulų kolonos bioreaktoriai	14
1.2. Maišymo greičio ir sparnuotės formos įtaka masės perdavimo reikšmei	15
1.3 k_{La} nustatymas naudojant dirbtinį neuronų tinklą	21
1.3.1. Dinaminis modelis	22
1.3.3 Deguonies-dujų balanso metodas	23
1.3.4 Anglies dioksido dujų balanso metodas	24
1.3.5 Duomenų suderinimas per prognozuojantį neuronų tinklą (arba tiesioginės sklaidos dirbtinį neuronų tinklą)	24
2. METODINĖ DALIS	29
2.1 Masės perdavimo procesai bioreaktorių sistemose	29
2.1.1 Masės perdavimas sistemoje „Skystis – kietas kūnas“	29
2.1.2 Masės perdavimas sistemoje „Skystis – skystis“	30
2.1.3 Masės perdavimas sistemoje „Dujos – skystis“	31
2.1.4 Deguonies masės balanso metodas	32
2.2 Dinaminių sistemų modeliavimui ir valdymui skirtas dirbtinis neuronų tinklas, NNSYSID paketas	33
2.3 EKSPERIMENTINIAI TYRIMAI	37
2.3.1 I Procesas	37
2.3.2 II Procesas	37
2.3.3 kLa apskaičiavimas naudojant stacionarios būsenos metodą	38
2.3.4 kLa reikšmių modeliavimas ir įvertinimas naudojant dirbtinį neuronų tinklą	39
3. TYRIMO REZULTATAI	42
3.1 kLa reikšmių mokymas ir testavimas naudojant Matlab ir NNSYSID, Levenberg-Marquardt algoritmą	42
3.2 kLa prognozavimas per neuronų tinklą, naudojant Matlab ir Neutral network tool	49
IŠVADOS	58
Literatūra:	59
Priedai	61
Priedas Nr.1. I proceso DNT tinklo mokymo ir testavimo duomenų vidutinės kvadratinės paklaidos naudojant Matlab ir NNSYSID	61

Priedas Nr.2. II proceso DNT tinklo mokymo ir testavimo duomenų vidutinės kvadratinės paklaidos naudojant Matlab ir NNSYSID	62
Priedas Nr.3. I ir II proceso DNT tinklo testavimo duomenų vidutinės kvadratinės paklaidos naudojant Matlab ir Nntool.....	63
Priedas Nr.4. Matlab programa.....	64

SANTRUMPŲ IR ŽENKLŲ AIŠKINIMO ŽODYNAS

- $k_L a$ – masės perdavimo koeficientas, [h^{-1}];
- DNT – dirbtinis neuronų tinklas;
- OUR – deguonies sunaudojimo greitis;
- RT – Rushton turbina;
- P4B – 4 įžambių menčių turbina;
- P2B – 2 įžambių menčių turbina;
- CD-6 – įgaubtų menčių diskinė turbina;
- OTR – deguonies tiekimo greitis;
- $C_L^* - C_{AG}^*$ – deguonies koncentracija, [$kmol/m^3$];
- $C_L - C_{AG}$ – deguonies koncentracija skysčio terpėje, [$kmol/m^3$];
- P_g – maišymo galia, [W];
- V – bioreaktoriaus darbinis tūris, [m^3];
- V_g^{β} – burbulų tūris reaktoriuje, [m^3];
- N^3 – maišymo greitis, [rpm];
- D^2 – sparnuotės diametras, [m];
- CMCNa – karboksimetriceliuliozės sodos druskos tirpalas;
- STB – (Stired tank bioreactor) bioreaktorius su maišykle;
- RPB – (Reciprocating plate boreactor) grįžtamai slenkamojo judesio bioreaktorius;
- RQ – kvėpavimo koeficientas;
- pO^2 – deguonies dalinis slėgis [atm].;
- N_A – deguonies srautas [$kmol m^{-2} s^{-1}$];
- y_{ei} – deguonies masės balansu apskaičiuotos $k_L a$ reikšmės;
- y_{ai} – neuronų tinklo apmokymo metu gautos $k_L a$ reikšmės;
- N – iteracijų skaičius;
- M – vidurkis;
- S – standartinis nuokrypis;
- DO – ištirpęs deguonis.

ĮVADAS

Temos aktualumas

Efektyvus masės perdavimo procesas yra esminis veiksnys vykti bet kokiai cheminei arba mikrobinei reakcijai bioreaktorių sistemose. Koeficientas k_La yra svarbiausias technologinis parametras valdant, reguliuojant aerobinę fermentaciją. k_La pagalba galima įvertinti aeracijos efektyvumą ir kiekybiškai išreikšti deguonies tiekimo operacijos kintamuosius.

Masės perdavimo procesas bioreaktorių sistemose svarbus projektuojant ar palyginant bioreaktorių efektyvumą, bioreaktorių maišymo įrangos efektyvumą, o taip pat valdant mikroorganizmų kultūros augimo ir produkto sintezės procesą.

Aerobiniuose bioprocusuose deguonies perdavimas yra itin svarbus, bet koks žymus deguonies sumažėjimas paveikia biocheminio proceso eigą. Todėl bioreaktorių sistemose ypatingas dėmesys skiriamas masės perdavimo procesams.

Pastaruoju metu labai paplitęs dirbtinių neuroninių tinklų taikymas įvairiems uždaviniams spręsti. Šiame darbe pritaikytas šiuolaikinis tiesioginės sklaidos dirbtinis neuronų tinklas, parodoma, kaip, naudojant DNT modelius, įvertinti k_La priklausomybę nuo šio parametro vertę įtakančių valdymo/proceso parametrų.

Principinis proceso matematinis modelis naudojamas reikiamiems duomenų rinkiniams apskaičiuoti, kad būtų atliktas sukurto dirbtinio neuronų tinklo mokymas ir patikrinimas.

Sukurti modeliai gali būti naudojami atliekant modeliu pagrįstą bioprocreso optimizavimą, kai masės perdavimo procesų intensyvumas yra vienas iš proceso našumą ribojančių veiksnių.

Darbo tikslas ir uždaviniai

Darbo tikslas – ištirti deguonies perdavimą pateiktuose biotechnologiniuose procesuose ir sukurti dirbtinių neuroninių tinklų modelius, gebančius prognozuoti masės perdavimo iš dujinės į skystąją fazę koeficiento k_La kitimą, atsižvelgiant į proceso ir valdymo kintamųjų vertes.

Darbo uždaviniai – apžvelgti masės perdavimo procesus bioreaktorių sistemose, bioreaktorių ir biotechnologinių procesų tipus. Išsiaiškinti tiesiogiai matuojamus ir netiesiogiai įvertinamus proceso kintamuosius ir technologinius parametrus, darančius įtaką masės perdavimo procesui. Apžvelgti šiuo metu masės perdavimo procesams (masės perdavimo iš dujinės į skystąją fazę koeficientui k_La modeliuoti/įvertinti) taikomus matematinius modelius, nurodyti jų privalumus ir trūkumus. Atlikti konkrečių pateiktų biotechnologinių procesų realių duomenų analizę ir parengiamuosius skaičiavimus. Sukurti ir realizuoti masės perdavimo koeficiento k_La netiesioginio įvertinimo modelius, pasitelkiant įvairios struktūros juodosios dėžės (dirbtinių neuroninių tinklų) modelius Matlab programavimo aplinkoje, identifikuoti jų

parametrus pateiktiems procesams, ištirti modelių užtikrinamą prognozavimo kokybę. Palyginti įvairios struktūros modeliais gautus rezultatus. Pateikti susistemintus tyrimų rezultatus ir išvadas apie sukurtų modelių panaudojimo galimybes.

1. APŽVALGINĖ DALIS

1.1 Bioreaktoriai

Bioreaktorius yra svarbus įrenginys bet kokiuose pramoniniuose arba laboratoriniuose biocheminiuose procesuose. Bioreaktoriaus veikimo principas priklauso nuo daugelio veiksnių:

- biomasės koncentracijos;
- sterilių sąlygų;
- efektyvaus maišymo;
- aušinimo;
- pamaitinimo medžiagų tiekimo;
- produkto pašalinimo;
- produkto sulaikymo;
- aeracijos;
- medžiagų metabolizmo.

Šiuo metu pramonėje naudojami trijų grupių bioreaktoriai:

1. nemaišomos, neaeruojamos sistemos: apie 70% bioreaktorių priklauso šiai kategorijai;
2. nemaišomos, bet aeruojamos sistemos: joms priklauso apie 10% bioreaktorių;
3. maišomos ir aeruojamos sistemos: joms priklauso apytiksliai 20% pramoninių bioreaktorių.

Nemaišomi, aeruojami bioreaktoriai naudojami tradiciniams gamybos procesams: vyno, alaus ir sūrio gamybai. Daugumai naujai atrastų bioprocesų naudojamos aeruojamos ir maišomos sistemos.

Pagrindinė tinkamai suprojektuoto bioreaktoriaus funkcija yra sukurti valdomą aplinką tam, kad būtų galima pasiekti optimalų augimą ir produkto formavimą tam tikroje ląstelių sistemoje. Bioreaktorių veikimas priklauso nuo daugelio funkcijų, įskaitant šias:

- biomasės koncentracija turi išlikti aukšta ir pasiekti aukštą produkto kiekį;
- turi būti sudaromos sterilios sąlygos kultūrų sistemų švarumui užtikrinti;
- tolygiam substratų ir mikroorganizmų pasiskirstymui bioreaktoriuje būtinas efektyvus maišymas;
- reikalinga šilumos cirkuliacija tam, kad bioreaktorius veiktų pastovioje temperatūroje, kuri reikalinga optimaliam mikroorganizmų augimui.

Bioreaktoriai skirstomi pagal tai, ar juose vykstančiuose procesuose naudojamas deguonis, kaip jis tiekiamas ir maišomas, pagal tai, ar biomasė yra suspenduota, ar imobilizuota,

koku režimu į bioreaktorių tiekama žaliava, substratai, ar pakartotinai grąžinama dalis kultivavimo terpės arba biomasės, ar proceso metu ištraukiama kultivavimo terpės dalis.

Atsižvelgiant į gaminamą produktą arba atliekamą funkciją, gamybinio masto reaktorių dydis gali svyruoti nuo kelių šimtų litrų iki dešimčių tūkstančių kubinių metrų.

Taip pat skiriasi juose matuojami ir valdomi technologiniai parametrai. Pagrindiniai jų yra temperatūra, pH, slėgis, aeracijos srautas, ištirpusių ir iš reaktoriaus ištekantių dujų koncentracija, optinis terpės tankis ir kt. [1]

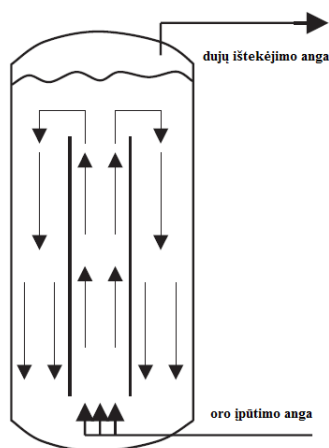
Bioreaktoriai klasifikuojami į:

- Anaerobinius ir aerobinius bioreaktorius;
- Nenutrūkstamo veikimo bioreaktorius;
- Periodinio veikimo bioreaktorius;
- Pusiau periodinio veikimo sistemas;
- Bioreaktorius su suspenduota mikroorganizmų masė.

1.1.1 Aerliftiniai bioreaktoriai

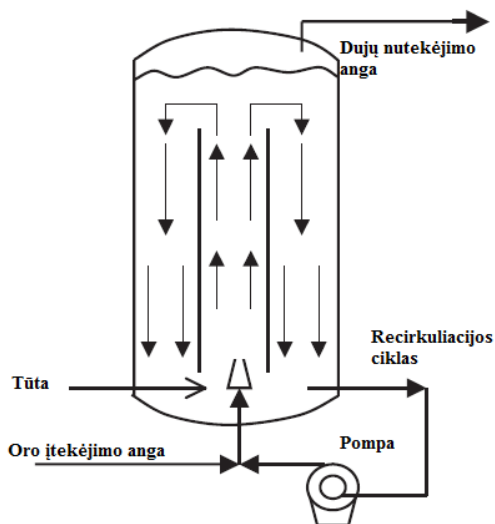
Aerliftiniame reaktoriuje maišymas vykdomas be jokio mechaninio maišymo. Aerliftiniai bioreaktoriai naudojami kultūroms, kurios yra jautrios pažeidimams ir normalus maišymas negalimas. Taikomi įvairių tipų aerliftiniai reaktoriai. Plačiausiai naudojami aerliftiniai bioreaktoriai – su vidiniu ir išoriniu ciklu.

Bioreaktorius su vidiniu ciklu – tai ilgas cilindrinis reaktorius, į kurį įstatomas kitas cilindras atvirais galais, reaktoriaus apačioje yra oro įpurškimo sistema, oro burbuliukai kyla cilindru ir sukelia skysčio srautą cilindro viduje iš apačios į viršų, o cilindro išorėje – iš viršaus žemyn [1].



1.1 pav. Reaktoriaus su vidiniu ciklu schema [5]

Bioreaktoriuose su išoriniu kontūru – oras per bioreaktorių transportuojamas pumpavimu. Juose naudojamas išorinis ciklas su mechaniniu pumpavimu. Dujos ir cirkuliuojantis skystis yra įpurškiami į cilindrą pro tūtą. 1.2 pav. pavaizduotas aerliftinis bioreaktorius, dirbantis su išoriniu recirkuliacijos siurbliu.



1.2 pav. Oro-keliamasis bioreaktorius su išoriniu recirkuliacijos siurbliu [5]

Aerliftinis reaktorius turi šiuos pranašumus:

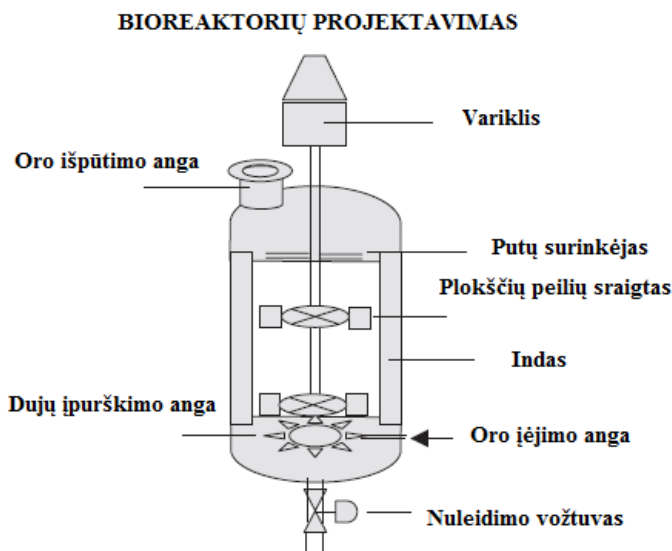
1. Bioreaktorius gali būti naudojamas augalinių ir gyvulinių ląstelių auginimui;
2. Kadangi nėra maišymo, išlaikomas sterilumas;
3. Didelėse talpose skysčio aukštis gali būti didesnis kaip 60 m, slėgis talpos apačioje padidina deguonies tirpumą, todėl padidėja $k_L a$ reikšmė;
4. Gali būti konstruojamos maksimaliai didelės talpos. Pavyzdžiui, genetiškai vienalyčio baltymo gamyboje suminis reaktoriaus tūris yra $2300 m^3$ (7 m skersmens ir 60 m aukščio vamzdis su $1560 m^3$ darbinio reaktoriaus tūriu). Šiame reaktoriuje buvo auginami mikroorganizmai, dėl biocheminių reakcijų išsiskyrė didelis šilumos kiekis. Tokio aukšto egzoterminės reakcijos šilumos kiekio neįmanoma pašalinti su įprastais maišomais reaktoriais.

Pagrindiniai aerliftinių reaktorių trūkumai yra tokie:

1. Aukšta didelių reaktorių kaina;
2. Aukšta energijos kaina, nes reikia prapūsti daugiau oro, oras turi būti didesnio slėgio, ypač didelėse talpose;
3. Kai mikroorganizmai cirkuliuoja bioreaktoriuje, keičiasi sąlygos ir neįmanoma išlaikyti reikiamą pastovų, maistingųjų medžiagų ir deguonies kiekį bioreaktoriuje;
4. Dujų atskyrimas nuo skysčio neefektyvus esant putojimui.

1.1.2 Bioreaktoriai su maišyklėmis

Svarbiausi pramonei bioreaktoriai yra įprasti su maišomomis talpomis, kurie turi dvigubą pranašumą: yra santykinai pigūs ir jų eksploatacija pigi. 1.3 pav. yra pavaizduota tokio reaktoriaus konstrukcinė schema.



1.3 pav. Bioreaktoriai su maišykle [5]

Šio tipo bioreaktoriai turi mechaninio maišymo sistemas, sudarytas iš variklio ir rotoriaus. Mechaninis maišymas paspartina deguonies tirpimo procesą ir mitybinių medžiagų bei mikroorganizmų masės homogenizavimą visame reaktoriaus tūryje. Jei reaktorius yra aerobinis, tai maišyklės užduotis – disperguoti orą į smuklius burbuliukus, taip padidinti dujų ir skysčio sąlyčio paviršiaus plotą ir palengvinti deguonies perdavimą iš dujinės į skystą fazę. Geriausiomis savybėmis pasižymi diskinės turbinos, tačiau jos pasižymi didelėmis energijos sąnaudomis. Siekiant sumažinti šias sąnaudas, reaktoriuje gali būti įrengtos kelios mažesnės diskinės maišyklės arba orui tiekti naudojamas kompresorius ir oro įpurškimo sistemos. Projektuojant svarbu atsižvelgti į bioreaktoriaus efektyvumą, nes deguonies įterpimas į skystį gali sudaryti reikšmingą proceso sąnaudų dalį [1].

Bioreaktoriuose su maišyklėmis, dėl didelio greičio ir aeracijos, atsiranda problemų, tokių, kaip putojimas. Putojimą slopinančios priemonės ne visada gali būti taikomos, nes jos gali turėti mikroorganizmų augimo stabdymo efektą. Kaip sprendimas, stabdantis putojimą, gali būti taikomas paprasčiausias įrenginys „žarstiklis“, sumontuotas ant maišančiojo veleno ir įrengtas skysčio paviršiuje. Kokybiškesnis arba sėkmingesnis įrenginys yra „Funda – putojimo sistema“, kurioje putos yra panaikinamos išcentrinėmis jėgomis. Maistingi tirpalai, esantys putose, nuteka atgal į bioreaktorių, o išlaisvintas iš putų oras palieka bioreaktorių.

Didelis dėmesys turi būti kreipiamas maišyklės konstrukcijai bioreaktoriaus viduje, nes ji lemia bioreaktoriaus darbą.

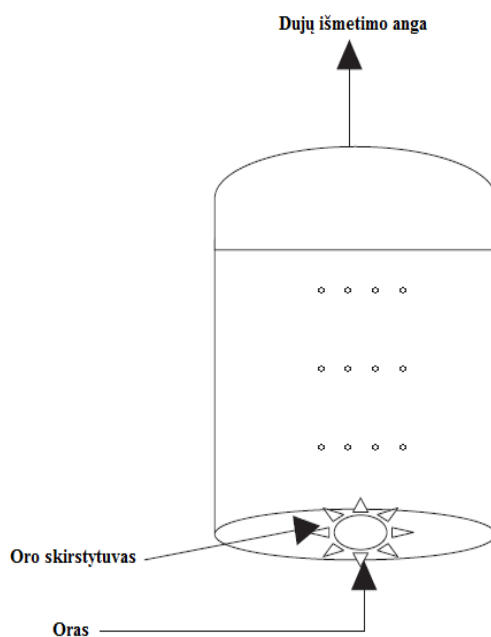
Įprasčiausias naudojamas maišytuvas yra keturių menčių diskinė turbina. Tačiau sistemų hidrodinamikos tyrimai parodė, kad diskinė turbina su 12 – 18 įgaubtų menčių turi tam tikrų pranašumų.

Vykdomi platūs tyrimai dujų-skysčio sistemose, kuriose problema yra ląstelių paviršiaus pažeidimas. Biosistemoms, kurios yra jautrios pažeidimams, ir turi kietų kūnų, pranašesnės yra turbinos su įlinkusiais peiliukais.

1.1.3 Burbulų kolonos bioreaktoriai

Mielių, alaus ir acto gamybai naudojami burbulų stulpo reaktoriai. Jie taip pat naudojami nutekamųjų vandenų aeracijai ir apdirbimui. Projektuojant tokį bioreaktorių, skysčio indo aukščio ir skersmens santykis ($H:D$) yra apie 2:1, kai bendruoju atveju šis santykis yra 3:1; mielių gamyboje santykis $H:D$ yra 6:1. Esant burbulų kolonom, masės perdavimo hidrodinamika priklauso nuo burbulų dydžio bei nuo to, kaip jie išsilaisvina iš oro skirstytuvo arba aeratoriaus.

1.4 pav. pavaizduotas nesudėtingas cilindrinis indas su purkštuvu, instaliuotu cilindro dugne, kuris įleidžia orą, praeinanti pro skystį.

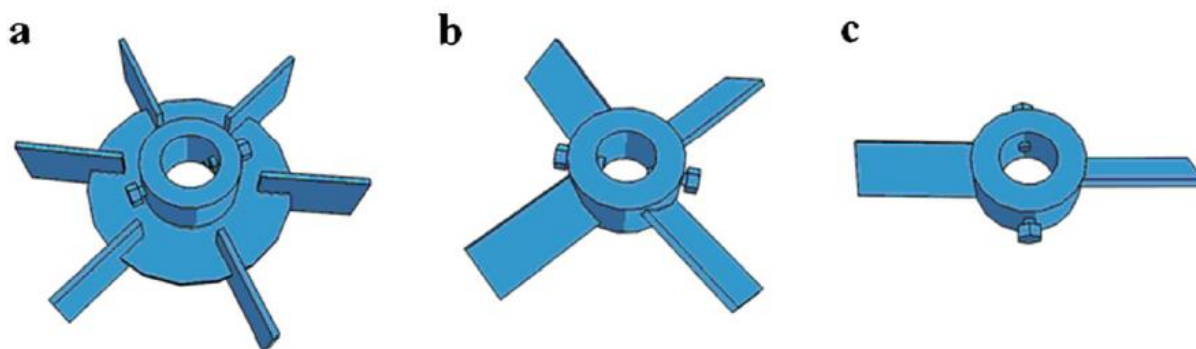


1.4 pav. Burbulų kolonos bioreaktoriai (cilindrinis bioreaktoriai) [5]

1.2. Maišymo greičio ir sparnuotės formos įtaka masės perdavimo reikšmei

Maišomuose bioreaktoriuose, deguonies masės perdavimas yra daugelio kintamųjų, tokių, kaip fizikinės skysčio savybės, klampumas, paviršiaus įtempimai, talpos ir maišytuvo forma, purkštovo tipas ir operavimo sąlygos, funkcija. Deja, literatūroje pateikiama informacija apie šią tikslią kintamųjų įtaką masės perdavimui yra nepakankama.

Maišomuose bioreaktoriuose užtikrinamos didelės masės ir šilumos perdavimo greičių reikšmės ir puikios maišymo sąlygos. Šiose sistemose didelis kintamųjų skaičius veikia masės perdavimą ir maišymą, bet svarbiausi iš jų yra maišytuvo greitis, maišytuvų tipas ir skaičius bei dujų srauto greitis. Svarbiausias vaidmuo atitenka sparnuotei. Ji atlieka tris pagrindinius uždavinius: kietų kūnų suspendavimą, maišymą ir reikiamo kiekio deguonies perdavimą į skystąją fazę bei tarpfazinio ploto maksimizavimą tarp dujų ir skysčio fazių. Labiausiai ištirtinėtoms sparnuotėms yra standartinės Rushton turbinos, skirtingo žingsnio menčių turbinos ir įvairių mentelių ir jų kombinacijos iš dviejų ir trijų mentelių.



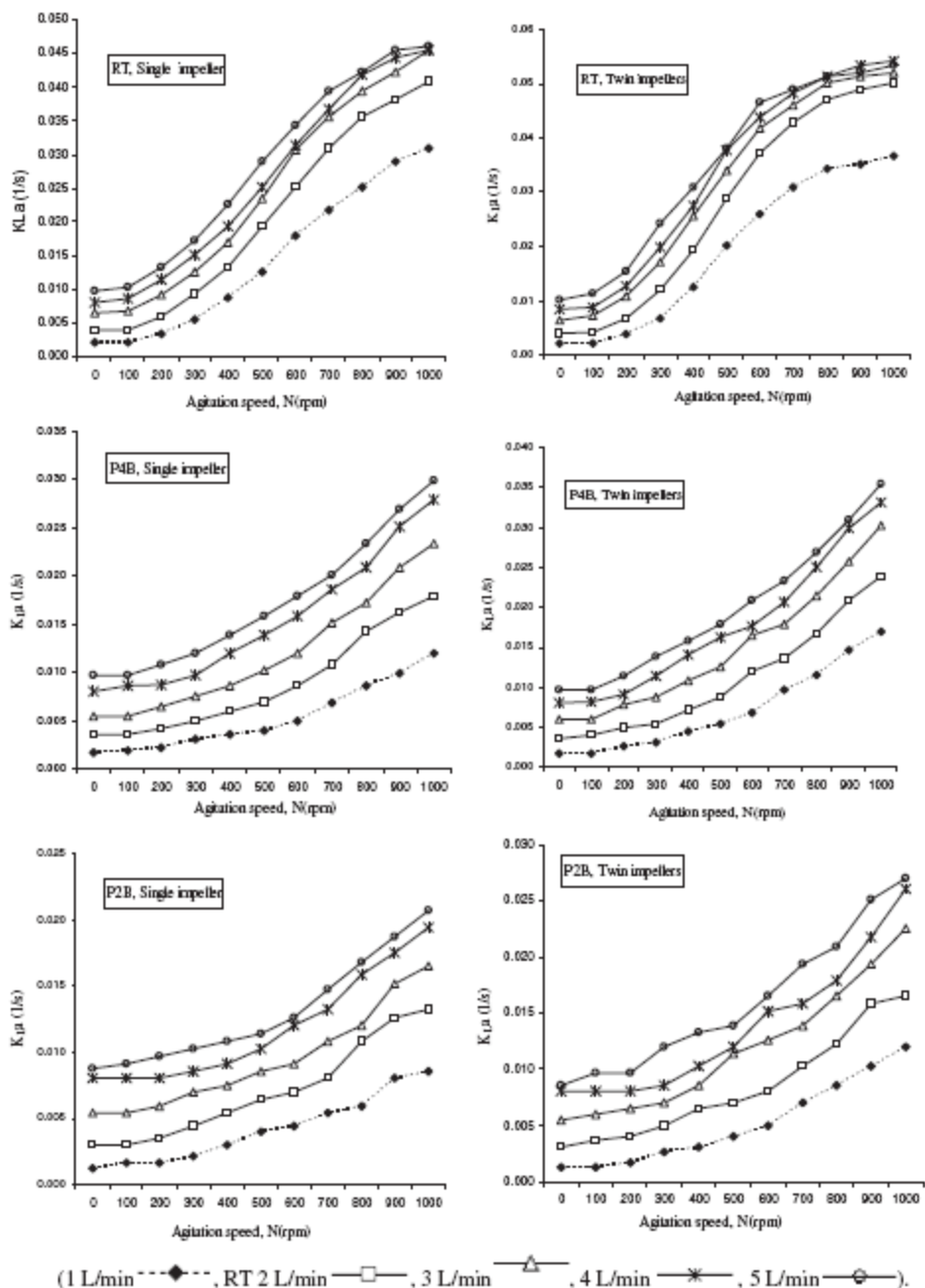
1.5 pav. a) Rushton turbina (RT), b) 4 įžambių menčių turbina (P4B), c) 2 įžambių menčių turbina (P2B) [14]

Tam, kad būtų galima optimizuoti sparnuočių projektavimą arba komponavimą efektyviai dujų dispersijai, svarbu suprasti geresnio deguonies masės perdavimo mechanizmą.

Ali Karimi [14] ištyrė deguonies masės perdavimo charakteristikas įvairioms vieno ir dviejų sparnuočių sistemoms. Buvo naudojamos trijų tipų sparnuotės, kaip parodyta 1.5 pav.

Tam, kad būtų galima įvertinti optimalias bioreaktoriaus sąlygas, buvo testuojama deguonies masės perdavimo greičio įtaka keliuose veikimo režimuose, apimant skirtingus sparnuočių tipus ir purkštuvus bei įvairius maišymo ir aeracijos greičius.

1.6 pav. iliustruoja rezultatus, kurie parodo, jog sparnuotės, kurių pagalba gautos didesnės $k_L a$ reikšmės, yra RT tipo. $k_L a$ reikšmės yra veikiamos proceso kintamųjų, tokių, kaip sparnuočių ir aeracijos greitis.



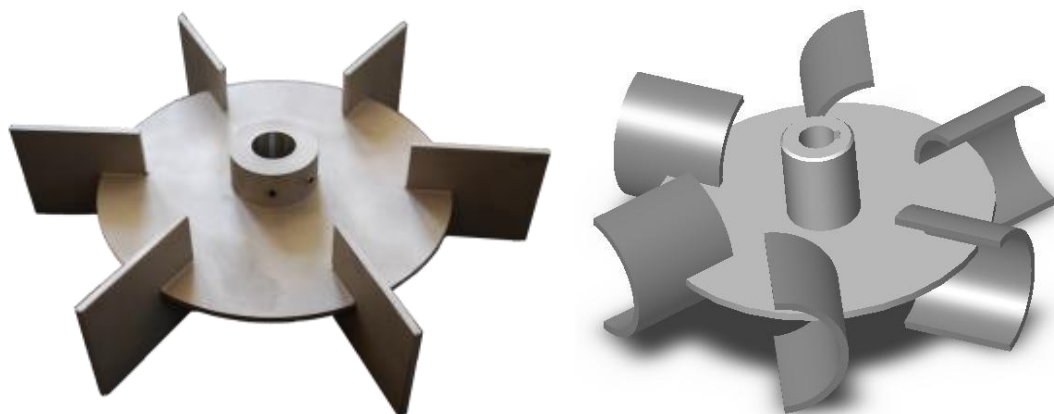
1.6 pav. Sparnuočių tipo įtaka k_{La} reikšmei [14]

RT tipo sparnuotės padidina k_{La} reikšmes vidutiniškai nuo 50% iki 69% ir nuo 60% iki 77%, lyginant su P4B ir P2B tipo sparnuotėmis. RT tipas yra efektyvesnis skaidant oro burbulus, kadangi turi didesnę skerspjūvio plotą. Didėjant paviršiniam burbulų plotui, didėja deguonies perdavimo greitis.

Rushton turbina ilgiau nei šešis dešimtmečius yra žinoma kaip plokščių menčių sparnuotė. RT – žinoma kaip generuojanti radialinį srauto lauką ir geba sudaryti postūmio sąlygas, reikalingas deguoniui įveikti dujų-skysčio plėvelės pasipriešinimą. Vis dėlto ši sparnuočių sistema turi ir keletą trūkumų, t.y., mažos dujų manipuliavimo galimybės, susidaro dujiniai lašai dėl didelio dujų kavitacijos efekto. Dujų kavitacija susidaro dėl žemo slėgio zonų,

sukurtų už besisukančių menčių išorės, ir akumuliuoja plūduriuojančius ant paviršiaus dujų burbuliukus. Taip pat reikia nepamiršti, kad, esant labai dideliame dujų srauto greičiui, gali pakilti skysčio lygis reaktoriuje.

RT geometrinių formų evoliucijos tendencijos, siekiant pagerinti dujų dispersijos charakteristikas, pirmiausia kito Smith pastangomis nuo 1980 metų, vėliau tobulinimus tęsė keletas britų mokslininkų ir pagrindinis atliktas patobulinimas buvo originalus – 90° kampu vieną į kitą pakrypusias menteles pakeitė mentės su skirtingu kreivumo laipsniu ir atsirado taip vadinama „Įgaubtų menčių diskinė turbina“ (CD-6) 1.7 pav. b. Atlikta įgaubtos formos menčių inovacija pakeitė dujų kavitacijos kontūrą maišymo greičio pagalba. Paskelbta, kad CD-6 minimizavo RT trūkumus, padidinusi dujų kavitacijos matmenis ir sužadinsi dujų lašų energiją, kurių pagalba pagerėjo manipuliavimo vykdymas. Be to, tiek RT, tiek CD-6 išpurškia skystį radialine kryptimi. Dviejų sparnuočių atveju daroma prielaida, kad du radialinio srauto laukai maišo ir dujas išsklaido efektyviau negu viena sparnuotės konfigūracija.



1.7 pav. Rushton turbina (RT), įgaubtų menčių diskinė turbina (CD-6). [8]

Nors literatūroje jau yra paskelbta daug tyrimų apie masės perdavimo koreliacijas naudojant įvairaus tipo sparnuotes, tačiau vis dar trūksta informacijos, kuri pateiktų palyginamąją sparnuočių analizę, susidedančių iš RT ir CD-6 turbinų, ir abiejų turbinų derinio darbą vienoje maišymo sistemoje.

Kelių sparnuočių naudojimo metodas yra vertinamas kaip vienas efektyviausių metodų, siekiant pagerinti deguonies difuzijos greitį maišomo mišinio bioreaktoriaus viduje. Kaip reikšmingas privalumas, atsirandantis iš operavimo keliomis sparnuotėmis, yra didelė energijos sklaida. Efektyvesnė dvifazė dujos-skystis cirkuliacija talpoje, esant mažesniai maišymo greičiui, eliminuoja nedidelį šlyties deformacijų efektą sparnuotėje.

Hipotezei įrodyti buvo atliktas Nurashikin Suhail [13] tyrimas lyginant 1.7 pav. tipo sparnuotes. Aerobinės fermentacijos procesuose ypatingas dėmesys sukonzentruotas dujų-skysčio masės perdavime. Deguonies perdavimo greitis OTR maišomo mišinio bioreaktoriuje yra išreiškiamas tokia lygtimi (1.1):

$$OTR = k_L a (C_L^* - C_L) \quad (1.1)$$

$k_L a$ reikšmė maišomame bioreaktoriuje priklauso nuo talpos vidinės ir sparnuočių geometrinių formų, fiziko-cheminių ir reologinių žadinamo skysčio savybių, maišymo intensyvumo ir aeracijos greičio. Šių narių tarpusavio priklausomybė tipiškai išreiškiama plačiai rekomenduojama koreliacija:

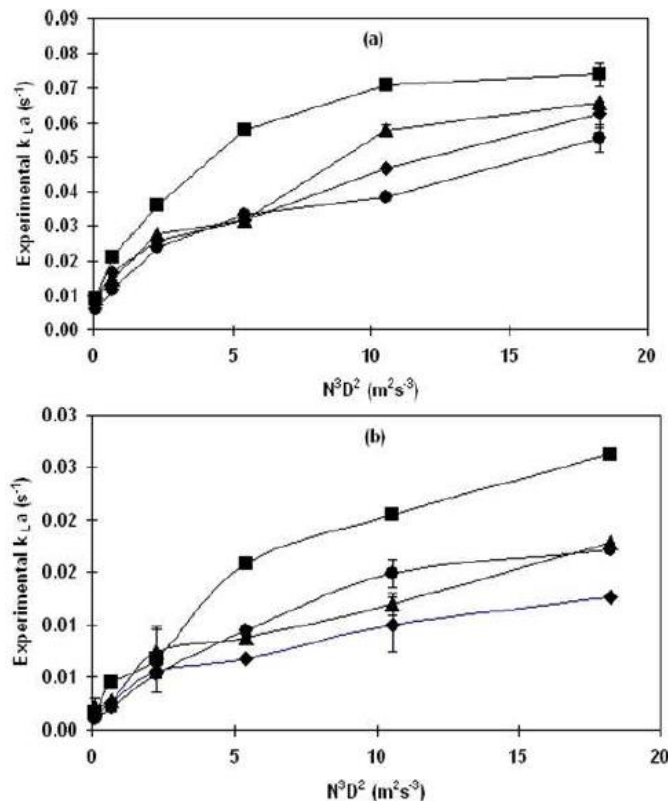
$$k_L a = C_l \left(\frac{P_g}{V} \right)^\alpha V_g^\beta \quad (1.2)$$

Nurashikin Suhail [13] tyrime, tiriant bedimensinės sparnuotės galios skaičiaus santykį, buvo išvesta lygtis, siejanti P_g/V su $(N^3 D^2)$, remiantis jų tiesiškumu ($y=x+1$). Tuo tikslu lygtis buvo pertvarkyta į tokią:

$$k_L a = C_l (N^3 D^2)^\alpha V_g^\beta \quad (1.3)$$

N yra maišymo greitis ir D yra sparnuotės diametras (m). Šis metodas naudingas, kaip eliminuojantis fluktuacijas ir matavimo netikslumus, matuojant maišymo galią įėjime.

$\frac{P_g}{V}$ pakeitimas $N^3 D^2$ leidžia nustatyti $k_L a$ priklausomybę nuo maišymo greičio ir sparnuotės formos. 1.8 pav. pavaizduota tendencija skirtingų konfigūracijų RT ir CD-6 distiliuoto vandens sistemoje ir 0,35% (w/v) CMCNa skysčio tirpalo sistemoje.



1.8 pav. a) Niutono skysčio sistema, b) Ne Niutono skysčio sistema. [13]

Paveiksle proporcingai pavaizduoti sąryšiai tarp $k_L a$ ir maišymo greičio, šios tendencijos pavaizduotos visoms sparnuočių konfigūracijoms ir abiejose skysčių sistemose. Šios tarpusavio priklausomybės priežastis yra ta, kad didėjant maišymo greičiui, išauga energijos

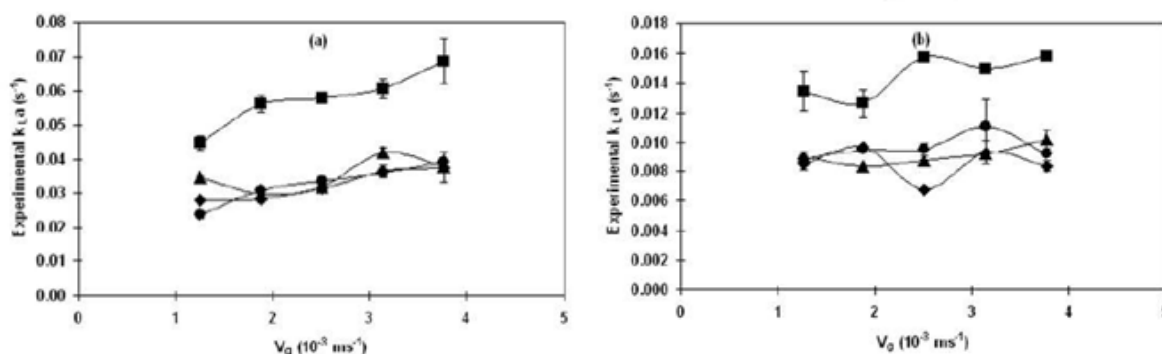
disipacija, kuri padidina burbulų sproginėjimo efektyvumą ir padidina deguonies absorbcijos į skystį greitį.

Įprastai k_{La} bei $N^3 D^2$ funkcijų pavidalai visose skysčio tekėjimo greičiuose yra daugiau ar mažiau panašūs. Distiliuoto vandens sistemoje dvigubos turbinos CD-6 gautas didesnis k_{La} reikšmių diapazonas, pranašumas, lyginant su kitomis sparnuočių konfigūracijomis, yra apie 30%. Be to, dviguba CD-6 sistema parodo, kad gaunamas nuo 5% iki 49% didesnis deguonies perdavimo greitis nei įprastai.

Panašūs Nurashikin Suhail [13] pastebėjimai buvo nustatyti ne-Newtoninių skysčių atveju, kur buvo naudojamas 0,35% (w/v) CMCNa skysčio tirpalas. CD6 – CD6 sparnuočių kombinacija dominavo prieš kitas, kur didesnis dujų-skysčių masės perdavimas pastebėtas daugelyje testuotų oro tėkmės greičių. Tuo tarpu RT ir CD6 kombinacijoje vienoje maišymo sistemoje dirba šiek tiek geriau nei du RT, tačiau skirtumas yra nereikšmingas ir naudingas tik palyginimo tikslais.

Didesnis masės perdavimas, gaunamas CD-6 sistemoje, yra dėl pagerintų fizikinių savybių. CD-6 pavaizdavo geresnę dujų dispersiją, kai skystis pumpuojamas žemyn ir tai visiškai prieštarauja RT, kuris pumpuoja priešinga srautui kryptimi, t.y., aukštyn. Skysčio judėjimas žemyn, veikiant CD-6, įgalina geresnį burbulų pasiskirstymą apatinėse indo dalyse. Sukombinavus du CD-6 ant to paties veleno, kaip ir buvo tikėtasi, sujungta konfigūracija reikšmingai pagerino dujų-skysčio masės perdavimo vykdymą ir tai gali paaiškinti dvigubos CD-6 sistemos pranašumą prieš kitas konfigūracijas.

Ryšys tarp k_{La} ir V_g charakterizuoja aeracijos greičio įtaką dujų-skysčio masės perdavimui. Aeracijos greičio įtaka k_{La} skirtingoms mentelių konfigūracijoms pavaizduota 1.9 pav.



1.9 pav. a) Niutono skysčio sistema, b) Ne Niutono skysčio sistema. [13]

Dvigubos CD-6 sistemos dominavimas prieš kitą sparnuočių konfigūraciją rodo, kad abiejose skysčių sistemose vidutinis procentinis skirtumas, lyginant su kitomis sparnuočių konfigūracijomis, yra daugiau negu 30%.

Nurashikin Suhail [13] teigimu, matematinė išraiška visoms konfigūracijoms pademonstravo tiesinę k_{La} priklausomybę nuo oro srauto greičio abiemis testuojamiems skysčiams. Kai maišymo metu įpučiamas oras praeina pro sparnuotės sritį, kurioje yra didžiausia turbulencija visame inde, vyksta burbulų susidūrimas, to pasekmėje susiformuoja mažesni burbulai, padarydami skysčių-dujų tarpfazinį plotą platesniu ir tuo pačiu padidėja deguonies perdavimo greitis. Burbulų sproginėjimo greitis labai priklauso nuo maišymo greičio ir dispersijos, kurią suteikia naudojama sparnuotė. Mažesni burbuliukai gaunami didesniame turbulentiškume ir esant didesniems tangentiniams įtempimams sparnuotės zonoje. Tačiau literatūroje taip pat teigiama, jog naudojant CD-6 gaunami mažesni tangentiniai įtempimai negu RT.

1.3 k_{La} nustatymas naudojant dirbtinį neuronų tinklą

Deguonies masės perdavimo koeficientas k_{La} yra svarbiausias parametras valdant arba reguliuojant aerobinę fermentaciją. k_{La} naudojamas lyginant bioreaktorių efektyvumą, bioreaktorių maišymo įrangos efektyvumą, o taip pat valdant mikroorganizmų kultūros augimo ir produkto sintezės procesą. Naudojami keturi pagrindiniai metodai deguonies masės perdavimo koeficientui k_{La} įvertinti: dinaminis metodas; stacionarus metodas, paremtas išankstiniu deguonies sunaudojimo greičio OUR nustatymu; deguonies balanso dujose metodas; anglies dioksido balanso metodas. Kiekvienu metodu gaunamos skirtingos k_{La} reikšmės. Kita alternatyva, daug greitesniam k_{La} nustatymui, yra dirbtinių neuronų tinklų taikymas, skirtas identifikuoti, prognozuoti k_{La} reikšmę.

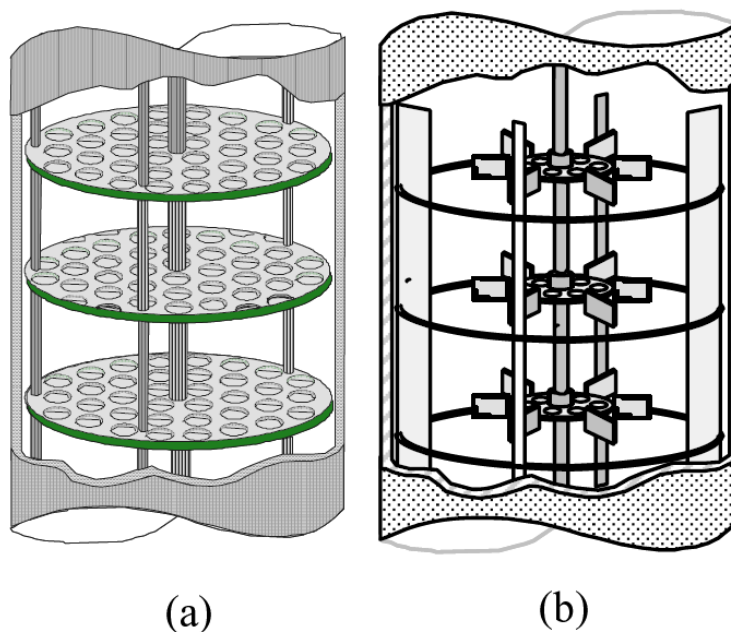
Literatūroje pateikiama daug metodų k_{La} reikšmei nustatyti. Dauguma tyrinėjimų, siekiant atkartoti kiek įmanoma tikslesnes fermentacijos sąlygas, buvo atlikti su vandeniu ir kitokiais skysčiais. Šie tyrinėjimai yra labai naudingi, kadangi sąlygos teisingai apibrėžtos ir gali būti griežtai kontroliuojamos bei gautas santykinai geras k_{La} įvertis, kuris gali būti naudojamas projektiniuose skaičiavimuose. Tačiau svarbu k_{La} nustatyti proceso metu, esant tikroms perdavimo sąlygoms, kadangi mikroorganizmai, substratai, metabolitai, klampios priemonės prieš putojimą turi įtakos deguonies perdavimui. Faktas, kad k_{La} reikšmės reaktoriuose gali žymiai skirtis nuo reikšmių, prognozuotų deguonies absorbcijai į vandenį, net jeigu atsižvelgta į skysčio fizikines savybes, tokias kaip klampumas ir difuziškumas.

Praktikoje taikomi du bendriausi metodai k_{La} reikšmei nustatyti – tai dinaminis metodas ir dujų ir deguonies masės balanso metodas. Naudojant šių dviejų metodų informaciją, galima lengvai apibrėžti kitus du metodus: stacionarų metodą ir anglies dioksido gamybos greičio metodą. Dinaminiam metodui, o vėliau ir stacionariam metodui, reikia sparčiai iširti ištirpusio deguonies kiekį rezultatams gauti. Du dujiniai metodai, kurie reikalauja deguonies ir anglies dioksido jutiklių, naudojami globaliam deguonies ir anglies dioksido balansui dujų fazėje bioreaktoriuje.

k_{La} akivaizdžiai nepriklauso nuo metodo, taikomo jo reikšmės apskaičiavimui. Skaičiuojant keturiais skirtingais metodais gaunamos keturios skirtingos k_{La} reikšmės dėl matavimo paklaidų, masės balanso lygčių netikslumo.

Yra atlikta tyrimų, kuriuose pasiūlyta įvertinti k_{La} reikšmę, naudojant tiesioginės sklaidos neuronų tinklą, kuris mokomas naudojant teorinius masės balanso modelius. Tyrimuose dažniausiai naudojamas masių balanso modelis apima dinaminį metodą, stacionarų metodą ir du dujų balanso metodus.

Nilesh Patel ir Jules Thibault [10] tyrime aptartas duomenų suderinimo metodas $k_L a$ nustatymui, naudojant dirbtinį neuronų tinklą. Jų palyginimas su įprastaisiais duomenų suderinimo metodais buvo atliktas su trimis mikroorganizmais: *Saccharomyces cerevisiae*, *Aspergillus niger* ir *Trichoderma reesei*. Šie trys mikroorganizmai buvo naudojami skirtingu laiku taip, kad fermentacijos sąlygos ir matavimų įranga būtų skirtinga.



1.10 pav. Konstrukcinė (a) grįžtamai slenkamojo judesio ir (b) maišomo bioreaktorių schema [10]

Kaip parodyta 1.10 pav., tyrime buvo naudojami dviejų skirtingų tipų bioreaktoriai: maišomas bioreaktorius (STB) ir grįžtamai slenkamojo judesio bioreaktorius (RPB). Šie bioreaktoriai yra identiški, išskyrus maišymo mechanizmus. STB buvo naudojamas trijų tipų mikroorganizmų fermentacijai, o RPB buvo naudojamas fermentacijoms su *Aspergillus niger*.

1.3.1. Dinaminis modelis

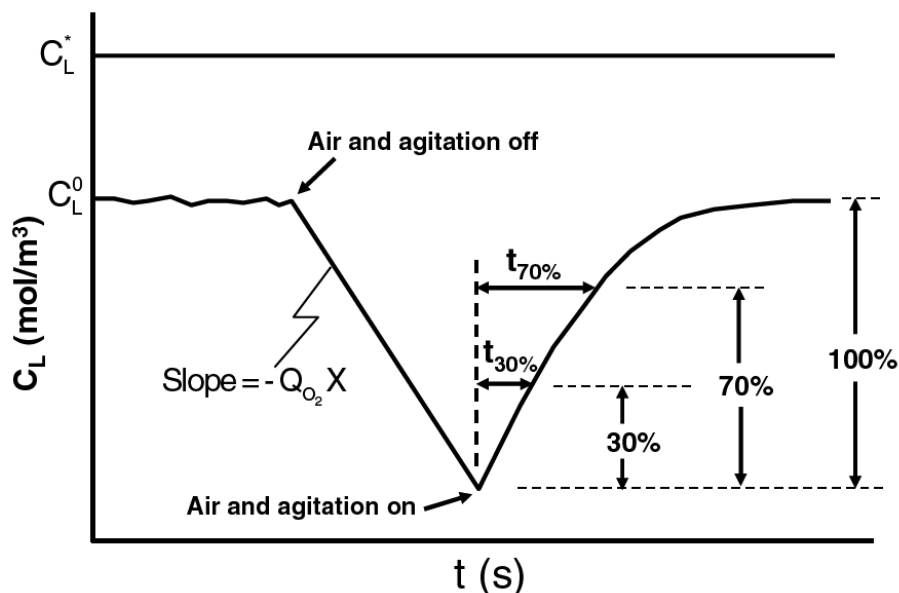
Dinaminis deguonies masės balansas yra išreiškiamas tokia formule:

$$\frac{C_L}{dt} = k_L a (C_L^* - C_L) - OUR \quad 1.4$$

(1.4) lygtis parodo, kad ištirpusio deguonies kitimo greitis bioreaktoriuje yra lygus deguonies masės perdavimo greičiui iš dujinės į skystąją fazę minus deguonies sunaudojimo greitis.

Dinaminiame metode deguonies sunaudojimo greitis OUR ir $k_L a$ nustatomi naudojant procedūrą, kaip pavaizduota 1.11 pav., dujų tiekimas ir maišymas yra nutraukiami akimirksniu tam, kad būtų nutrauktas deguonies tiekimas į skystąją fazę taip arba tiek, kad ištirpusio deguonies mažėjimo greitis būtų sąlygotas vien tik OUR . Ištirpusio deguonies kiekis paprastai mažėja pagal tiesinį dėsnį ir tiesės krypties koeficientas C_L , kaip laiko funkcija, tiesiogiai

įvertina deguonies sunaudojimo greitį. Nulinė hipotezė teigia, kad deguonies sunaudojimo greitis neveikiamas oro burbulų ir maišymo bei žemesnės ištirpusio deguonies koncentracijos. Prieš pasiekdama žemiausią ištirpusio deguonies ribą, ištirpusio deguonies koncentracija, aeracija ir maišymas yra atnaujinami ir ištirpusio deguonies koncentracijoje grįžta į pradinį lygį. $k_L a$ gali būti apskaičiuotas pagal (1.4) lygtį, naudojant grafinį metodą arba sprendžiant lygtį baigtinių skirtumų metodu.



1.11 pav. Ištirpusio deguonies reakcijos grafikas [10]

$k_L a$ įvertinimas, taikant dinaminį metodą, nepriklauso nuo OUR įvertinimo. Kaip matoma, pagal lygtį (1.4), OUR veikia tik stacionarią ištirpusio deguonies reikšmę ir neveikia greičio. Tokiu būdu $k_L a$ įvertinimas, naudojant ištirpusio deguonies atsaką į mišinio masės aprūpinimą deguonimi, nepriklauso nuo tikros OUR reikmės.

1.3.2. Stacionarus metodas

Kai OUR reikšmė yra žinoma, taikant (1.5) lygtį ir turint pseudo stacionarios fazės duomenis, galima tiesiogiai apskaičiuoti $k_L a$:

$$k_L a = \frac{OUR}{C_L^* - C_L^0}; \quad (1.5)$$

1.3.3 Deguonies-dujų balanso metodas

Esant pseudo stacionarioms sąlygoms, deguonies tiekimo greitis yra lygus deguonies sunaudojimo greičiui OUR, todėl OUR (1.5) lygtyje gali būti pakeistas dujinio deguonies bendros masės balansu tam, kad būtų galima įvertinti $k_L a$:

$$k_L a = \frac{\frac{1}{V_L} \left(\frac{P_1}{RT_1} Q_{1,G} y_{1,O_2} - \frac{P_2}{RT_2} Q_{2,G} y_{2,O_2} \right)}{(C_L^* - C_L^0)}; \quad (1.6)$$

1.3.4 Anglies dioksido dujų balanso metodas

Dujinio anglies dioksido išsiskyrimo greitis taip pat gali būti naudojamas įvertinti $k_L a$. Tai įmanoma, jeigu yra žinomas kvėpavimo koeficientas (RQ). $k_L a$ apskaičiuojamas taip:

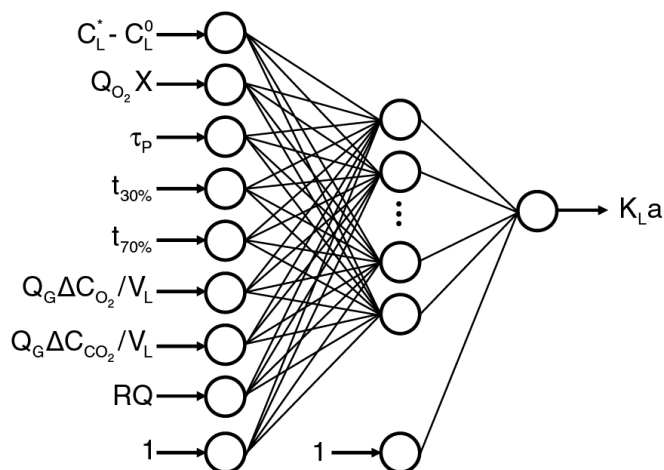
$$k_L a = \frac{\frac{1}{RQ} \frac{1}{V_L} \left(\frac{P_2}{RT_2} Q_{2,G} y_{2,CO_2} - \frac{P_1}{RT_1} Q_{1,G} y_{1,CO_2} \right)}{(C_L^* - C_L^0)}; \quad (1.7)$$

Visiems šiems metodams būdingi tam tikri privalumai ir trūkumai skirtingose fermentacijos procesų fazėse. Pavyzdžiui, dinaminis metodas gali būti taikomas su santykinai aukštu tikslumu pradinėse fermentacijos fazėse, kai deguonies sunaudojimas yra žemas, o ištirpusio deguonies koncentracija medžiagoje yra didelė.

Dėl mažo deguonies sunaudojimo, fermentacijos stadijoje skirtumas tarp įeinančių ir išeinančių dujų koncentracijų yra labai mažas. Dujų balanso metodas $k_L a$ įvertinimui stokoja tikslumo. Kai fermentacija ima vykti intensyviau, biomasės augimas tuo pačiu didina deguonies sunaudojimą ir mažina ištirpusio deguonies koncentraciją. Dėl to dujų balanso metodas tampa tikslesnis.

1.3.5 Duomenų suderinimas per prognozuojantį neuronų tinklą (arba tiesioginės sklaidos dirbtinį neuronų tinklą)

Duomenų suderinimo (Data reconciliation) algoritmai reikalauja sudėtingų matematinių skaičiavimų, kurie yra taikomi visiems $k_L a$ vertinimams. Tam, kad būtų galima supaprastinti procedūrą, Nilesh Patel ir Jules Thibault [10] pasiūlė taikyti prognozuojantį neuronų tinklą DNT.



1.12 pav. Neuronų tinklas $k_L a$ įvertinimui [10]

Prognozuoti $k_L a$ proceso duomenys gali būti paprastai gaunami iš dinaminio eksperimento ir dujinio deguonies bei anglies dioksido koncentracijų. Prognozuojantis neuronų tinklas turi analogiją su žmogaus smegenų neuronine sistema, gebančia mokytis, modifikuodama sinopsines jungtis. Jį sudaro įėjimo neuronų sluoksnis, išėjimo neuronų sluoksnis ir neuronų skaičius paslėptame sluoksnyje.

Nilesh Patel ir Jules Thibault [10] tyrime buvo naudojamas trisluoksnis DNT. Tai tinklas su vienu paslėptu sluoksniu, kaip pavaizduota 1.12 pav. DNT galima paprastai išreikšti kaip apibendrintą netiesinį modelį, susijusį su nepriklausomų įėjimo kintamųjų rinkiniu ir priklausomų išėjimo kintamųjų rinkiniu. Neuronų tinklo įėjimai ir išėjimai paprastai yra normuoti nuo 0 iki 1. Įėjimo sluoksnio neuronai paprastai turi koduotą informaciją apie įėjimo kintamuosius ir perduoda tą informaciją paslėpto sluoksnio funkciniais neuronams. Paslėptame sluoksnyje yra funkciniai neuronai, kurie vykdo svorinių sumų netiesines transformacijas iš ankstesnio sluoksnio neuronų išėjime. Tam, kad adekvačiai atvaizduoti tarp sluoksnių fenomeną, kurį neuronų tinklas stengiasi sumodeliuoti, būtina treniruoti modelį su reprezentaciniu duomenų rinkiniu. Treniravimo fazėje taikomas optimizavimo algoritmas, kad surastų tinkamą parametrų rinkinį, jungtinius svorius, kurie minimizuos prognozavimo paklaidą visam duomenų treniruojamam rinkiniui.

1.1 lentelė. Modeliavimo kintamieji [10]

Kintamieji	Intervalas	Kintamieji	Intervalas
P (Pa)	99300 - 103300	$Q_G(10^4 m^3/s)$	1,00 – 1,67
T (K)	299 - 303	$y_{1,o2}$	0,1992 – 0,2092
$\tau_p(s)$	5 - 15	$y_{2,o2}$	0,189 – 0,205
$C_L^*(mol/m^3)$	0,23 – 0,25	$y_{1,CO2}(10^4)$	4,0 - 6,0
$C_L^0(mol/m^3)$	0,001 – 0,24	$y_{1,CO2}(10^4)$	5,72 – 229,3
$V_L(m^3)$	0,008 – 0,017	$Q_{O2X}(10^4)$	14,3 – 991,5
RQ	0,8, - 1,50	$K_L a(s)$	0,001 – 0,10

Atlikta 500 modeliavimų su visais proceso kintamaisiais. Šių kintamųjų variacijos diapazonas surašytas 1.1 lentelėje. Kiekvienam modeliavimui tiesiogiai buvo išrinkta arba nustatyta keletas proceso parametrų, naudotų neuronų tinklo įėjime k_{La} vertinimui. Šis parametrų rinkinys vėliau buvo naudojamas mokytį neuronų tinklą.

1.12 pav. pavaizduotas neuronų tinklas turi aštuonis proceso įėjimo kintamuosius, kurie buvo išrinkti, nes koreliuoja su k_{La} reikšme, vyraujančia bet kokiame fermentacijos taške.

1. $C_L^* - C_L^0$ yra skirtumas tarp ištirpusio deguonies (DO) koncentracijos ir pseudo-stacionari DO koncentracija prieš dinaminį testą.

2. Q_{O2X} arba OUR yra deguonies sunaudojimo greitis, nustatytas, kai maišymas ir aeracija sustabdomi. Ši reikšmė nedaro įtakos DO mėginio dinamikai, taigi, tiesinė kreivės atkarpa gali būti naudojama tiesioginiams vertinimams.

3. τ_p yra DO mėginio laiko pastovioji, kuri nustatyta atitinkamai pritaikant pirmos eilės modelį.

4. $t_{30\%}$ yra laikas, reikalingas ištirpusio deguonies koncentracijai nuo žemiausio taško grįžti iki 30% koncentracijos.

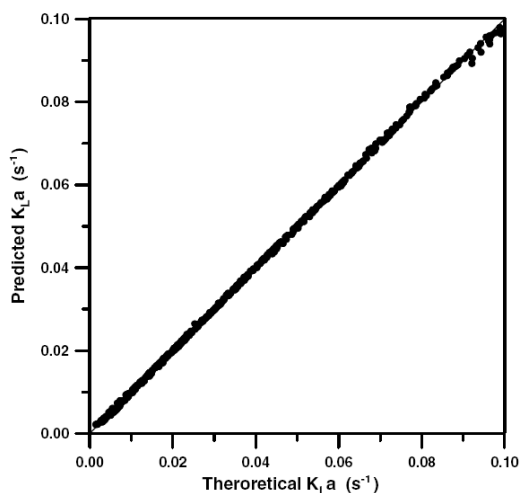
5. $t_{70\%}$ yra laikas, reikalingas ištirpusio deguonies koncentracijai nuo žemiausio taško grįžti iki 70% koncentracijos.

6. $Q_G \Delta C_{O_2}/V_L$ yra dujų srauto greičio sandauga ir deguonies koncentracija tarp fermenterio įėjimo ir išėjimo, padalintas iš skysčio tūrio.

7. $Q_G \Delta C_{CO_2}/V_L$ yra dujų srauto greičio sandauga ir anglies dioksido koncentracija tarp fermenterio įėjimo ir išėjimo, padalinta iš skysčio tūrio. Kai RQ reikšmė yra žinoma, šios dvi išraiškos yra idealiai ekvivalentiškos.

8. RQ yra kvėpavimo koeficientas ir yra konstanta pagrindinėje fermentacijoje.

Neuronų tinklas ištreniruotas, jis taikomas greitam k_{La} įvertinimui iš nustatytų 8 parametrų. Tinklas sudarytas iš 9 neuronų įėjimo sluoksnyje, 7 neuronų paslėptame sluoksnyje ir vieno neurono išėjimo sluoksnyje. Treniravimui taikomos kvazi-Niutono optimizavimo algoritmas. Palyginimui pateiktas grafikas 1.13 pav. teorinio k_{La} ir naudojant prognozuojantį neuronų tinklą. Kaip galima pastebėti, neuronų tinklas geba labai tiksliai atvaizduoti 500 k_{La} reikšmių, kurios buvo nustatytos naudojant plataus diapazono didelio skaičiaus proceso kintamųjų reikšmes.



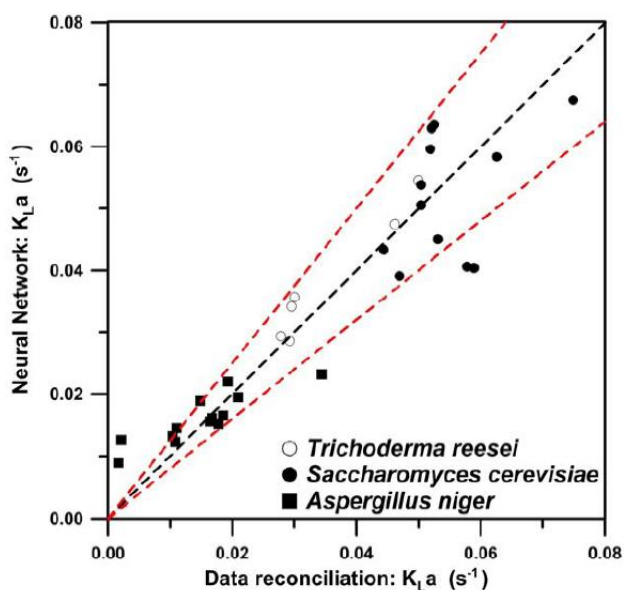
1.13pav. Teorinio k_{La} palyginamas su prognozuojamu k_{La} [10]

Neuronų tinklas mokytas su 400 reikšmių, parinktų atsitiktinai, o likusios 100 reikšmių buvo panaudotos patikrinimui.

Neuronų tinklo prognozės buvo palygintos su k_{La} reikšmėmis, gautomis įprastais duomenų apskaičiavimo metodais, rezultatai pavaizduoti 1.14 pav., k_{La} diapazonas trims fermentacijų sistemoms yra skirtingas. Vidutinis k_{La} vertinimų skirtumas tarp dviejų metodų yra trys brūkšninės linijos, vaizduojančios $\pm 20\%$ variaciją k_{La} prognozėje. k_{La} reikšmių skirtumų

Saccharomyces cerevisiae, *Aspergillus niger* išsibarstymas yra didesnis negu *Trichoderma reesei*. Autoriai teigia, kad šis skirtumas priklauso nuo įrangos, naudojamos matuoti dujų fazės koncentraciją, tikslumo. *Trichoderma reesei* matavimams buvo naudojamas masės spektrometras, kuris duoda žymiai didesnę tikslumą negu atlikus matavimus naudojant O_2 paramagnetinį ir CO_2 infraraudonųjų spindulių matavimą su kitais dviem mikroorganizmais.

Iš 1.14 pav. matome, kad rezultatai, gauti naudojant dirbtinį neuronų tinklą, yra panašūs, kaip ir gauti įprastais duomenų suderinamumo metodais.



1.14 pav. Teorinio k_{La} palyginamas su prognozuojamu k_{La} [10]

Taškai grafike 1.14 pav. matosi santykinai gerai pasiskirstę aplink 45 laipsnių tiesę ir nėra akivaizdžių sisteminių nukrypimų nė vienai iš trijų fermentacijų. Stebimas tikslumo lygis šiame palyginime yra santykinai geras ir jame gerai matomos tipinės k_{La} paklaidos. Rezultatai yra labai įdomūs, kadangi buvo panaudotas unikalus neuronų tinklas k_{La} įvertinimui trijų tipų mikroorganizmams.

Tikslumui padidinti, turi būti sudaryta galimybė naudoti treniravimo duomenis. Neuronų tinklo įėjimo vektorius gali būti sumažintas specifinių pritaikymų atveju. Pavyzdžiui, jeigu žinoma DO mėginio laiko pastovioji, jos reikšmė turi būti naudojama generuojant duomenis ir ši reikšmė turėtų būti pašalinta iš neuronų tinklo įėjimo vektoriaus. Taip pat yra naudinga atlikti koreliacinę analizę tarp eilės įėjimo duomenų ir k_{La} su treniravimo duomenimis, siekiant nustatyti turinčius didžiausią įtaką k_{La} įvertinimui, tuo būdu išrinkti geriausią įėjimo vektorių, tinkamą neuronų tinklui.

Priklausomai nuo veikimo režimų, gali būti naudojamos skirtingos neuronų tinklo versijos. Pavyzdžiui, fermentacijos pradžioje deguonis ir anglies dioksido koncentracijų skirtumas tarp įeinančių ir išeinančių dujų srautų yra mažas. Tikslus k_{La} įvertinimas šiais dviem metodais

griežtai priklauso nuo O_2 ir CO_2 sensorių ir masės srauto matuoklių. Tokiu atveju k_{La} reikšmė gali būti nustatyta labai tiksliai taikant dinaminį ir stacionarųjį metodus.

Specializuoti neuronų tinklai gali būti sukurti prognozuoti k_{La} reikšmę iš mažesnio duomenų rinkinio, kur ne visi masės tvermės modeliai gali būti naudojami. Tai turėtų padidinti k_{La} įvertinimo tikslumą. Taip pat rezultatai rodo, kad tipinė k_{La} reikšmės įvertinimo paklaida unikaliam dirbtiniam neuronų tinklui, pavaizduotam 1.12 pav., yra adekvati.

2. METODINĖ DALIS

2.1 Masės perdavimo procesai bioreaktorių sistemose

Konvekcinis masės perdavimas – tai masės perdavimas judančiame skystyje. Molekulinė difuzija vyksta koncentracijos gradiento mažėjimo kryptimi, tačiau pagrindinė skysčio masė taip pat juda, suminis masės perdavimo greitis yra didesnis dėl konvekcinių srautų susiliejimo. Masės perdavimo analizė yra svarbi daugiafazėse sistemose, kur besiliečiantys sluoksniai patiria reikšmingą masės perdavimo pasipriešinimą. Masės perdavimo greitis tiesiškai proporcingas perdavimo plotui ir perdavimo proceso varomajai jėgai:

$$\text{Perdavimo greitis} = \alpha \text{ perdavimo plotas} \times \text{varomoji jėga} \quad (2.1)$$

$$\text{Perdavimo greitis} = \text{masės perdavimo koeficientas} \times \text{perdavimo plotas} \times \text{varomoji jėga} \quad (2.2)$$

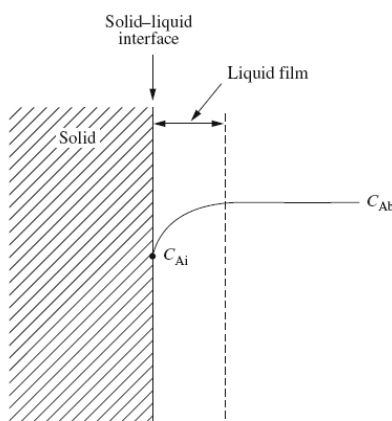
Kiekvienam skysčiui masės perdavimo varomoji jėga kraštiniame sluoksnyje gali būti išreikšta kaip koncentracijos skirtumas tarp skysčio plėvelių. Masės perdavimo greitis kraštiniame sluoksnyje tarp sluoksnių yra:

$$N_A = ka\Delta C_A = ka(C_{Ab} - C_{Ai}) \quad (2.3)$$

Bioprocusuose yra trys masės perdavimo sistemos: masės perdavimas sistemoje skystis – kietas kūnas, skystis – skystis tarp nenusimaišančių tirpalų ir dujos – skystis.

2.1.1 Masės perdavimas sistemoje „Skystis – kietas kūnas“

Masės perdavimas tarp judančio skysčio ir kieto kūno yra svarbus biologiniuose procesuose. Molekulių absorbcijai ant paviršių, tokių kaip, Chromatografija, reikalingas masės perdavimas iš skystos fazės į kietą. Masės perdavimas iš skystos į kietą kūną taip pat labai svarbus kristalizacijoje, kai molekulės juda iš skysčio ant augančio kristalo paviršiaus.



2.1 pav. Būsena tarp sluoksnyje Skystis – kietas kūnas [6]

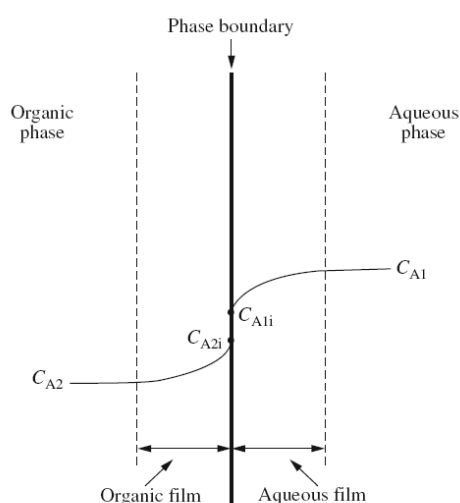
Jeigu **a** yra skystis - kietas kūnas tarp sluoksnyje, jo ploto vienetas lygus tūrio vienetai, tada masės perdavimo tūrinis debitas apskaičiuojamas taip:

$$N_A = k_L a (C_{Ab} - C_{Ai}) \quad (2.4)$$

kur k_L – skysčio fazės masės perdavimo koeficientas. Masės perdavimo koeficientas yra išmatuojamas arba apskaičiuojamas naudojant žinomas koreliacines funkcijas.

2.1.2 Masės perdavimas sistemoje „Skystis – skystis“

Masės perdavimas tarp nesusimaišančių tirpiklių yra dažnai sutinkamas bioprocesuose. Organiniai tirpikliai, naudojami atskirti antibiotikus, steroidus, dvifazės vandens sistemas, yra naudingi baltymo išgryninimui. Būsena tarpfluoksnyje tarp dviejų nesusimaišančių skysčių pavaizduota 1 pav.



2.2 pav. Būsena tarpfluoksnyje tarp dviejų nesusimaišančių skysčių [6]

Masės perdavimo greitis kiekvienoje skysčio fazėje apskaičiuojamas taip:

$$N_{A1} = K_{L1} a (C_{A1} - C_{A1i}) \quad (2.5)$$

$$N_{A2} = K_{L2} a (C_{A2i} - C_{A2}) \quad (2.6)$$

Lygčių pritaikymas yra sudėtingas, nes sunku išmatuoti skysčių koncentracijas tarpfluoksnyje. Dažniausiai yra priimta, kad koncentracijos yra pusiausvyros.

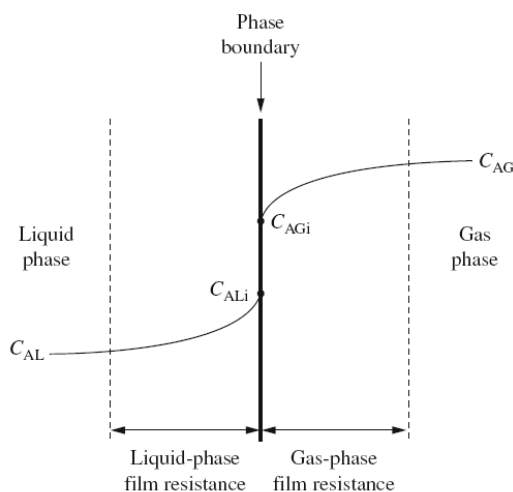
$$N_A = K_{L1} a (C_{A1} - m C_{A2}) \quad (2.7)$$

$$N_A = K_{L2} a \left(\frac{C_{A1}}{m} - C_{A2} \right) \quad (2.8)$$

Šios lygtys parodo, kad masės perdavimo greitis tarp dviejų skysčio fazių paprastai nepriklauso nuo koncentracijų skirtumo, tačiau pusiausvyra taip pat yra svarbus faktorius. Pagal (2.7) lygtį varomoji „A“ perdavimo jėga iš skysčio 1 į skystį 2 yra skirtumas tarp koncentracijos C_{A1} ir „A“ koncentracijos skystyje 1, kuris turėtų būti pusiausvyroje su koncentracija C_{A2} skystyje 2. Pagal (2.8) lygtį skirtumas tarp C_{A2} ir „A“ koncentracijos skystyje 2, turi būti pusiausvyroje su C_{A1} skystyje 1.

2.1.3 Masės perdavimas sistemoje „Dujos – skystis“

Masės perdavimas dujos – skystis sistemoje yra labai svarbus bioprocesuose dėl deguonies poreikio aerobinėse ląstelių struktūrose.



2.3 pav. Būsenų tarpfluksnyje sistemoje „Dujos – skystis“ [6]

Masės perdavimo greitis pro dujų ribinį sluoksnį yra:

$$N_{AG} = K_G a (C_{AG} - C_{AGi}) \quad (2.9)$$

Masės perdavimo greitis pro ribinį skysties sluoksnį yra:

$$N_{AL} = K_L a (C_{ALi} - C_{AL}) \quad (2.10)$$

Masės perdavimo greitį, kai koncentracijos yra pusiausvyros, galima išreikšti taip:

$$N_A = K_G a (C_{AG} - C_{AG}^*) \quad (2.11)$$

$$N_A = K_L a (C_{AL}^* - C_{AL}) \quad (2.12)$$

Tačiau ir šios lygtys gali būti supaprastintos sistemoms, kuriose pasipriešinimas masės perdavimui pagrinde yra tik dujų – skysties fazės tarpfluksnyje. Kai ištirpusi medžiaga „A“ yra labai tirpi skystyje, kaip amonio perdavimas į vandenį, skysties pusės pasipriešinimas yra mažesnis, palyginti su dujų fazės plėvele tarpfluksnyje. Iš lygties (2.13), jei skysties pusės pasipriešinimas mažas, $k_L a$ turi būti santykinai didesnis. Todėl, jeigu $k_G a$ yra apytiksliai lygus $K_G a$, gauname, kad:

$$N_A = k_G a (C_{AG} - C_{AG}^*) \quad (2.13)$$

Taip pat, jeigu „A“ yra mažai tirpus skystyje, kai deguonis vandens tirpale, skysties fazės masės perdavimo greitis dominuoja ir $k_G a$ yra daug didesnis už $k_L a$. Todėl, jeigu $k_G a$ yra apytiksliai lygus $K_G a$, lygtis supaprastinama:

$$N_A = k_L a (C_{AL}^* - C_{AL}) \quad (2.14)$$

Kadangi „dujų – skysties“ terpėse deguonies perdavimas vaidina pagrindinį vaidmenį daugelyje bioprocesų, todėl dažniausiai yra naudojama 2.14 formulė.

Skystis – skystis terpėje masės perdavimui įprastai yra labai sudėtinga apskaičiuoti tarpsluoksnio plotą „a“, kai dujos pernešamos pro skystį, tarpsluoksnio plotas priklauso nuo burbulų dydžio ir skaičiaus, kuris taip pat priklauso nuo daugelio faktorių, tokių kaip terpės sudėtis, maišymo greitis, dujų tiekimo greitis. Kadangi k_L taip pat yra veikiamas šių parametru, k_L ir a yra derinami kartu ir vadinami masės perdavimo koeficientu.

2.1.4 Deguonies masės balanso metodas

Deguonies masės balanso metodas yra patikima procedūra įvertinti $k_L a$, šis metodas leidžia nustatyti $k_L a$ iš „vieno taško“ matavimo. Svarbus privalumas, kad metodas gali būti taikomas proceso metu. Matavimo tikslumas stipriai priklauso nuo įeinančių ir išeinančių dujų sudėties, įėjimo ir išėjimo greičio, slėgio ir temperatūros. Tam, kad nustatyti $k_L a$, yra matuojama deguonies koncentracija dujų fazėje ir įtekantis bei ištekantis dujų srautas. Tūrinis masės perdavimo greitis skaičiuojamas taip:

$$N_A = \frac{1}{V_L} [(F_g C_{AG})_i - (F_g C_{AG})_0] \quad (2.15)$$

Kadangi dujų koncentracija paprastai matuojama kaip dalinis slėgis, todėl gali būti pritaikytas idealių dujų dėsnis ir formulė pertvarkyta taip:

$$N_A = \frac{1}{RV_L} \left[\left(\frac{F_g C_{AG}}{T} \right)_i - \left(\frac{F_g C_{AG}}{T} \right)_0 \right] \quad (2.16)$$

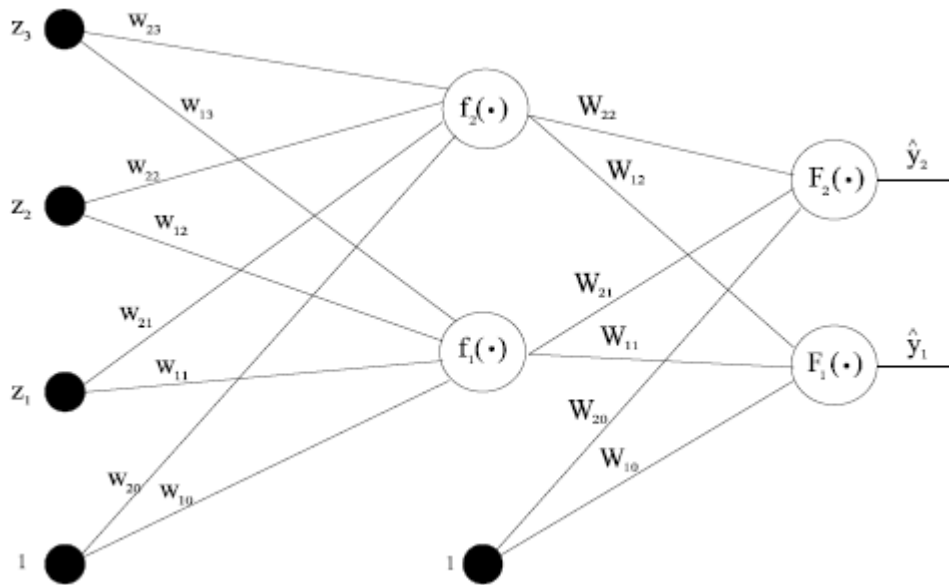
Tada $k_L a$ galima išreikšti taip:

$$k_L a = \frac{N_A}{(C_{AL}^* - C_{AL})} \quad (2.17)$$

Stacionariam metodui turi būti keletas prielaidų: skystoji fazė yra gerai išmaišyta; dujų fazė yra gerai išmaišyta.

2.2 Dinaminių sistemų modeliavimui ir valdymui skirtas dirbtinis neuronų tinklas, NNSYSID paketas

Daugiasluoksnis perseptronas (MLP – multilayer perceptron) ko gero yra dažniausiai nagrinėjamas neuronų tinklas. Pagrindinė priežastis – gebėjimas nesudėtingai modeliuoti labai sudėtingus funkcinius ryšius.



2.4pav. MLP neuronų tinklas su trimis įėjimais, dviem paslėptais neuronais ir dviem išėjimo neuronais [3]

Pavaizduotas neuronų tinklas su vienu paslėptu sluoksniu, hiperbolinio tangento ir tiesinėmis aktyvavimo funkcijomis (f, F):

$$\hat{y}_i(w, W) = F_i \left(\sum_{j=1}^q W_{ij} h_j(w) + W_{i0} \right) = F_i \left(\sum_{j=0}^q W_{ij} f_j \left(\sum_{l=1}^m w_{il} z_l + w_{j0} \right) + W_{i0} \right); \quad (2.18)$$

Svoriai, kurie apibrėžti vektoriumi θ arba alternatyviai – matricomis w ir W , yra reguliuojami tinklo parametrai ir nustatomi iš pavyzdžių rinkinio per procesą, vadinamą treniravimu. Pavyzdžiai arba treniravimui naudojami duomenys įprastai yra vadinami įėjimų rinkiniu $u(t)$ ir atitinkamai reikiamu rinkiniu $y(t)$.

Apibrėžiame treniravimo duomenų rinkinį taip:

$$Z^N = \{[u(t), y(t)] \mid t=1, \dots, N\}; \quad (2.19)$$

Treniravimo tikslo funkcija po to apibrėžiama kaip treniravimo duomenų rinkinio atvaizdas į galimų svorių rinkinį

$$Z^N \rightarrow \hat{\theta}; \quad (2.20)$$

taip, kad tinklas kurtų prognozes $\hat{y}(t)$, kurios, tam tikra prasme, yra „uždaros“ tikriems išėjimams $y(t)$.

Paklaidos prognozės metodas, kuris strategiškai taikomas naudojant šį paketą, yra pagrįstas vidutinės kvadratinės paklaidos kriterijumi:

$$V_N(\theta, Z^N) = \frac{1}{2N} \sum_{t=1}^N [y(t) - \hat{y}(t|\theta)]^T [y(t) - \hat{y}(t|\theta)]; \quad (2.21)$$

Svoriai gaunami taip:

$$\hat{\theta} = \arg \min V_N(\theta, Z^N); \quad (2.22)$$

taikant tam tikros rūšies iteracinę minimizavimo schemą:

$$\theta^{(i+1)} = \theta^{(i)} + \mu^{(i)} f^{(i)}; \quad (2.23)$$

$\theta^{(i)}$ apibrėžia esamą iteraciją (skaičiaus „i“), $f^{(i)}$ yra paieškos kryptis ir $\mu^{(i)}$ yra žingsnio dydis.

Egzistuoja didelis skaičius treniravimo algoritmų, kiekvienas iš jų yra charakterizuojamas pagal paieškos kryptį ir žingsnio dydį.

2.1 lentelė. Tinklų treniravimo algoritmai [3]

Tinklų treniravimo algoritmai	
batbp	Atvirkštinio paklaidos plitimo algoritmo programinė versija
igls	Iteracinis, apibendrintas mažiausių kvadratų tinklų treniravimo metodas, su daugiamačiais išėjimais
incbp	Rekursinis (prieauglio) atvirkštinės paklaidos metodas
marq	Levenberg-Marquardt metodas
marqlm	Atmintį taupantis Levenberg-Marquardt metodas
rpe	Rekursinis paklaidos prognozavimo metodas

Programos veikimui būtini šeši parametrai:

1. NetDef: apibrėžia tinklo architektūrą:

```
NetDef=[ 'HHHHHH'
         'LH- - - -']
```

reiškia, kad tinklas turi 6 hiperbolinio tangento neuronus paslėptame sluoksnyje, vieną tiesinį ir vieną hiperbolinio tangento išėjimą;

2. w_1, w_2 : pradinių svorių matricos (užduota pagal nutylėjimą programoje). Jeigu išreikšta laužtiniais skliaustais „[]“, parenkama atsitiktinė reikšmė.
3. PHI: įėjimo duomenų matrica.
4. Y: reikalingų išėjimo parametrų reikšmių matrica.
5. trparms: duomenų struktūra, turinti parametrus, susijusius su treniravimo algoritmu. Jeigu reikšmė neužduodama arba pažymima „[]“, tada bus naudojami programoje užduoti parametrai. Jeigu nenorima užduotų programiškai reikšmių, reikia naudoti funkciją SETTRAIN.

Pavyzdžiui, iškviečiama funkcija:

```
>> [W1, W2, crit_vector, iter]=batbp(NetDef, w1, w2, PHI,Y)
```

treniruos tinklą pagal gerai žinomą atvirktinio paklaidos plitimo algoritmą. Tai gradiento mažėjimo metodas, kuriuo atliekant skaičiavimus atsižvelgiama į specifinę neuronų tinklo struktūrą. Pridedant argumentą trparms

```
>> [W1, W2, crit_vector, iter]=batbp(NetDef, w1, w2, PHI,Y, trparms)
```

įmanoma pakeisti užduotas treniravimo programoje treniravimo algoritmo reikšmes. Kitose srityse trparms gali būti naudojama kartu su funkcija settrain:

```
>> trparms=stttrain; % pažymėti trparms kaip funkciją pagal nutylėjimą
```

```
>> trparms=stttrain(trparms, 'maxiter', 1000, 'critmin', 1e-4, 'eta', 0.02);
```

Pirmoji komanda užduoda kintamąjį trparms į duomenų struktūrą su nustatytais parametrais. Antroji komanda užduoda maksimalų iteracijų skaičių, lygų 1000, apibrėžia, kad treniravimas turi baigtis, jeigu įvertinimo funkcija V_N tampa mažesne už 10^{-4} , galiausiai užduoda žingsnio reikšmę, lygią 0.02. Funkcija batbp grąžins ištreniuotus svorius (W1, W2), vektorių, kuriame yra kiekvienos iteracijos parametro V_N reikšmės (crit_vector) ir iš tikro įvykdytų iteracijų skaičių (iter). Šis algoritmas šiuo metu yra populiariausias tinklų treniravimo metodas. Populiarumą nulėmė ne konvergavimo savybės, bet paprastumas, kuriuo pasiekiamas konvergavimas.

Levenberg-Marquardt metodas vadinamas standartiniu metodu, kuris minimizuoja pagal vidutinės kvadratinės paklaidos kriterijų. Plačiai naudojamas dėl greito konvergavimo ir stabilumo. Metodas įgyvendintas funkcijoje marq:

```
>> [W1, W2, crit_vector, iter, lambda]=marq(NetDef, w1, w2, PHI,Y, trparms)
```

Trparms funkcijos pagalba galima kontroliuoti treniravimą keletu naujų būdų. Trparms galima modifikuoti, kad būtų gautas prieauglio kriterijaus minimizavimas su sureguliuo nariu:

$$V_N(\theta, Z^N) = \frac{1}{2N} \sum_{t=1}^N [y(t) - \hat{y}(t|\theta)]^T (y(t) - \hat{y}(t|\theta)) \frac{1}{2N} \theta^T D \theta; \quad (2.24)$$

Funkcija settrain taip pat gali kontroliuoti pradinę λ reikšmę. Tai nėra svarbiausias kritinis parametras, kadangi jis sureguliuojamas adaptyviai ir tokiu būdu daroma įtaka tik pradiniam konvergavimo greičiui: jeigu jis per didelis, algoritmas sumažina žingsnį ir jeigu jis per mažas, algoritmas padidins jį iki tinkamos reikšmės.

marq priklauso taip vadinamų grupinio apdorojimo metodų klasei (reiškia „viskas iš karto“) ir iš karto okupuoja didelį kiekį atminties. Alternatyvi strategija yra pakartoti rekursinį algoritmą treniruojamiems duomenims tam tikrą skaičių kartų.

Kadangi svorinė matrica gali būti lengvai faktorizuojama naudojant MATLAB komandą sqrtm, tiesiausias būdas treniruoti tinklus su daugiamačiais išėjimais:

```
>>[W1,W2,crit_vector,iter,lambda]=marq(NetDef,w1,w2,PHI,sqrtm(inv(Gamma))*Y,trparms);
>> W2=sqrtm(Gamma)*W2;
```

Jeigu stebimas baltasis triukšmas ir pasiskirstymas yra Gauso bei tinklo architektūra yra pabaigta, t.y. architektūra yra pakankamai didelė aprašyti analizuojamus duomenis, taikomas maksimalaus tikėtinumų metodas gautų svorių įvertinimui, kai Λ yra parinktas kaip triukšmo kovariacijos matrica. Kovariacijos matrica daugeliu atvejų yra nežinoma ir todėl dažnai apskaičiuojama. Funkcija igls yra iteracinė procedūra tinklo treniravimui ir kovariacijos matricos apskaičiavimui.

Funkcijos išėjime yra normuoti svoriai ir tokiu būdu tinklo išėjimas galiausiai turi būti perskaičiuotas tinkamam masteliui.

Apibendrinant kiekvieno algoritmo pranašumus ir trūkumus, lentelėje pateikti svarbiausi mastelį apibūdinantys požymiai:

2.2 lentelė Algoritmo pranašumai ir trūkumai [3]

	Vykdyto laikas	Stabilumas	Mokymas	Atmintis
Batbp	--	+	--	-
Incbp	--	+	--	++
Marq	++	++	++	--
Marqlm	+	++	++	0
Rpe	0	-	--	++

2.3 EKSPERIMENTINIAI TYRIMAI

k_{La} apskaičiavimui gali būti naudojami šie mechanistiniai metodai: deguonies dinaminis, stacionarios būsenos; deguonies balanso dujose; anglies dioksido balanso. Norint taikyti šiuos metodus, būtina turėti sudėtingų matavimų duomenis. Ne visose biotechnologinėse sistemose yra dujų koncentracijos ir sunaudojimo, ir slėgio jutikliai, todėl, neturint pakankamai duomenų, reikiamų mechanistiniams modeliams, geriau remtis juodos dėžės modeliais.

Tyrimo DNT modeliai yra įgyvendinti naudojant kompiuterinę matematinę sistemą Matlab, versija R2013a, taip pat naudojant NNTOOL ir NNSYSID2.0 paketus. Naudoti DNT modeliai, kurių pagalba įvertinta k_{La} priklausomybė nuo šio parametro vertei darančių įtaką proceso parametrų, tokių kaip aeracija, maišymas, kultivavimo terpės tūris. Skaičiavimams naudojami dviejų realių procesų eksperimentiniai duomenys.

2.3.1 I Procesas

Pirmo proceso duomenys gauti iš optimizuoto rekombinuotų baltymų gamybos proceso. Pagrindinė bakterija šiame procese yra *E coli* BL21, išvesta iš pBR322 plazmido. Ji turi geną, užkoduotą kaip VP1 ir sujungtą su enzimo dehidratu (DHFR, EC 1.5.1.3). Papildomai turi geną, kuriame yra ampicilino. VPI-DHFR genas išreikštas per valdomą tac-promoter'į. Maišytuvą įrengtas su dvejomis standartinėmis T/2 Rushton turbinomis. Ląstelių koncentracija buvo nustatyta matuojant kultūrų absorbciją su 600 nm spektrometru (UV-2102PC) ir perskaičiuojant į sausos ląstelių masės koncentraciją su iš anksto užsiduota koreliacija. Proceso matavimams naudojami: analizės prietaisas YSI 2700, spektrometras UV-2102PC ir kiti prietaisai.

2.3.2 II Procesas

Antro proceso duomenys yra gauti iš rekombinuotų baltymų gamybos proceso. Kaip pagrindiniai sužadinimo organizmai, šiame procese buvo naudojamos *E. coli* bakterijos. *Escherichia coli* B, turinti plazmidą p UBS 520, naudojama kaip pagrindinis organizmas. Antras plazmidas p12023 išvestas iš dažnai naudojamo pKK223-3. Proceso tikslas – auginti biomasę didžiausiu augimo greičiu ir po tam tikro laiko kontroliuoti augimo greitį, manipuliuojant substrato padavimo greitį, kad iš anksto nustatytu galutiniu kultivavimo laiku produkto koncentracija pasiektų reikiamą reikšmę. Iš pradžių kultūra buvo auginama bioreaktoriuje gliukozės terpėje, iki kol gliukozė buvo išnaudota ir biomasės koncentracija pasiekė reikšmę apytiksliai $1 (g \cdot kg^{-1})$. Vėliau rekombinuoto baltymo gamyba buvo žadinama pridėdant laktozės, kuri vėliau buvo naudojama tolygiam ląstelės sužadinimui kaip C-šaltinis. Kultivacijos buvo vykdomos 10-l laboratorinio mastelio bioreaktoriuje.

Proceso matavimams naudojami: standartinis analizatorius (YSI 2700), smulkių dalelių analizatorius (Coulter Multisizer II), kultūros optinis tankis, išmatuotas tinkle tiesiogiai su 670

nm ilgio bangomis, taikant lazerį ASR LA – 300 LTm. Dujos buvo matuojamos dujų analizatoriumi, turinčiu paramagnetinį sensorių (Maihak OXOR 610) ir infraraudonuosius spindulius absorbuojančiu sensoriumi (Maihak UNOR 610), kuris matuoja dalinius slėgius O_2 ir CO_2 .

2.3.3 $k_L a$ apskaičiavimas naudojant stacionarios būsenos metodą

$k_L a$ nustatymui naudojamas deguonies masės balanso metodas:

$$\frac{C_L}{dt} = k_L a (C_L^* - C_L) - OUR; \quad (2.25)$$

Kadangi laiko momentu būseną stacionari, lygtį prisilyginame nuliui ir išsireiškiame $k_L a$:

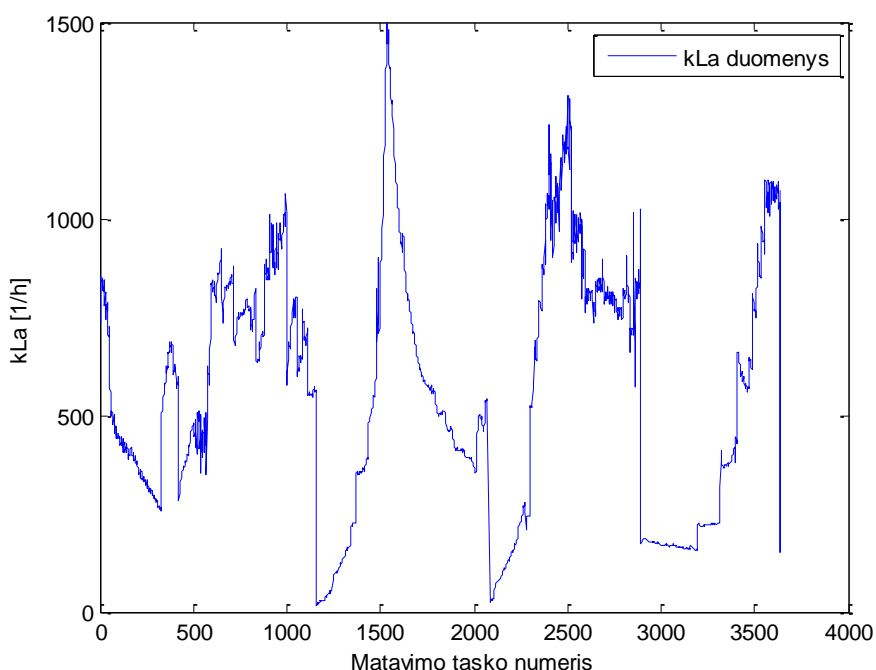
$$k_L a = \frac{OUR}{(C_L^* - C_L)}; \quad (2.26)$$

Atliekant tyrimą, deguonies tirpumas įvertintas $7,52 \times 10^{-3} [\text{kmol}/\text{m}^3]$ remiantis literatūra [6].

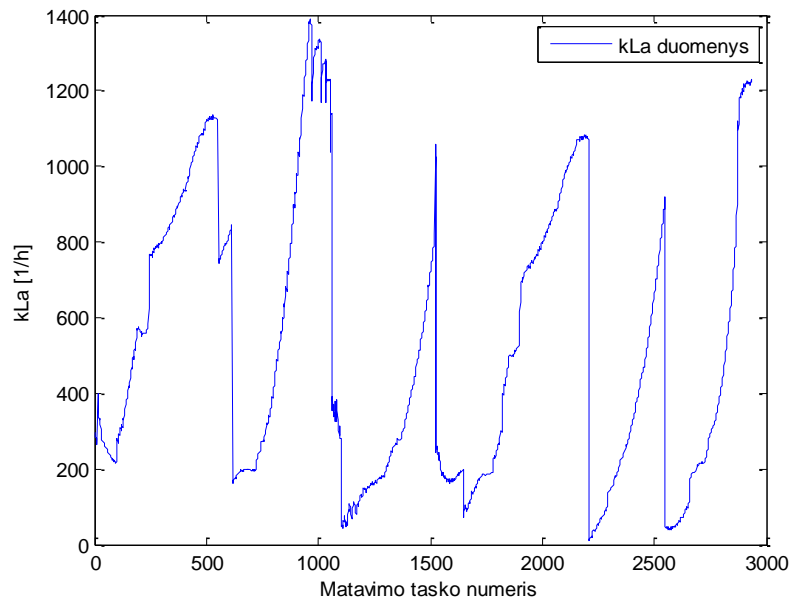
Ištirpusio deguonies koncentracija C_L apskaičiuota kaip parodyta formulėje 2.27, panaudojant dalinio slėgio pO_2 eksperimentinius proceso duomenis.

$$C_L = \frac{pO_2}{100}; \quad (2.27)$$

Žemiau pateiktos iš eksperimentinių duomenų apskaičiuotos $k_L a$ reikšmės:



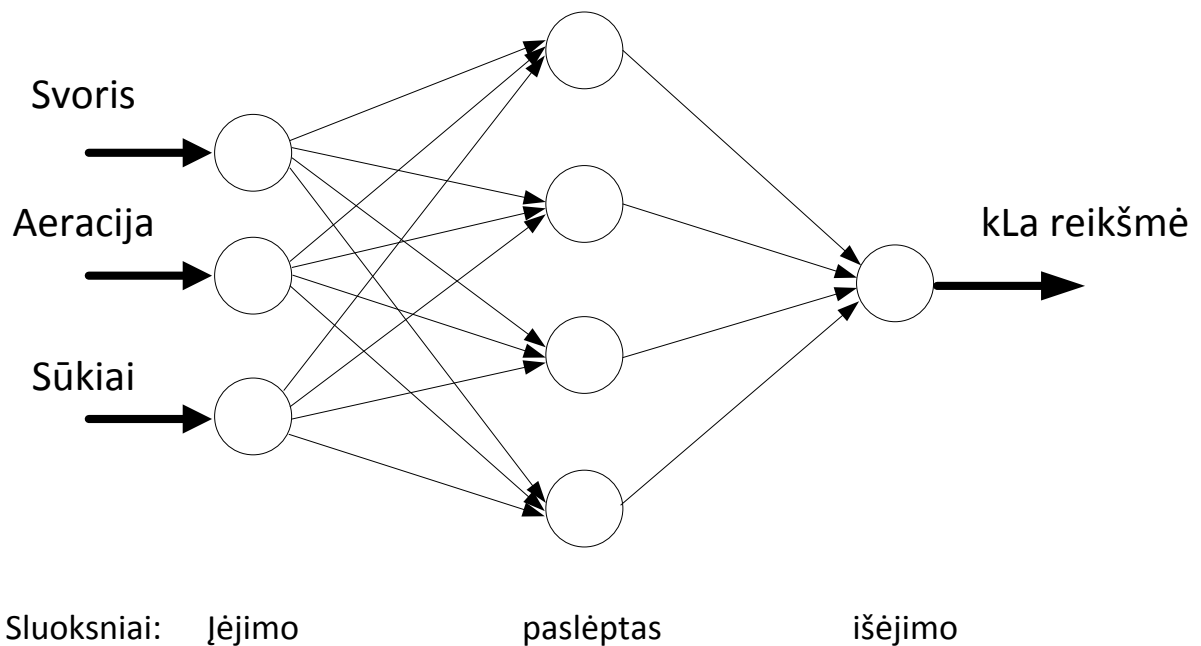
2.5 pav. I proceso eksperimentinių duomenų stacionarios būsenos metodu apskaičiuotos $k_L a$ reikšmės



2.6 pav. II proceso eksperimentinių duomenų stacionarios būsenos metodu apskaičiuotos $k_L a$ reikšmės

2.3.4 $k_L a$ reikšmių modeliavimas ir įvertinimas naudojant dirbtinį neuronų tinklą

Tyrimė dirbtinis neuronų tinklas pasirinktas su vienu paslėptu sluoksniu, neuronų tinklo įėjimo vektorius pasirinktas iš trijų kintamųjų – svorio, aeracijos ir sūkių – kadangi šiais kintamaisiais lengviausia manipuluoti proceso metu. Tyrimė naudotas neuronų tinklas pavaizduotas 2.7 pav.



2.7 pav. Dirbtinio neuronų tinklo schema

Neuronų tinklo treniravimui panaudoti deguonies masės balansu apskaičiuoti $k_L a$ duomenys, gauti iš dviejų visiškai skirtingų procesų, kuriuose naudojami skirtingi mikroorganizmai. Su šiais duomenimis neuronų tinklas ištreniruotas ir testuotas.

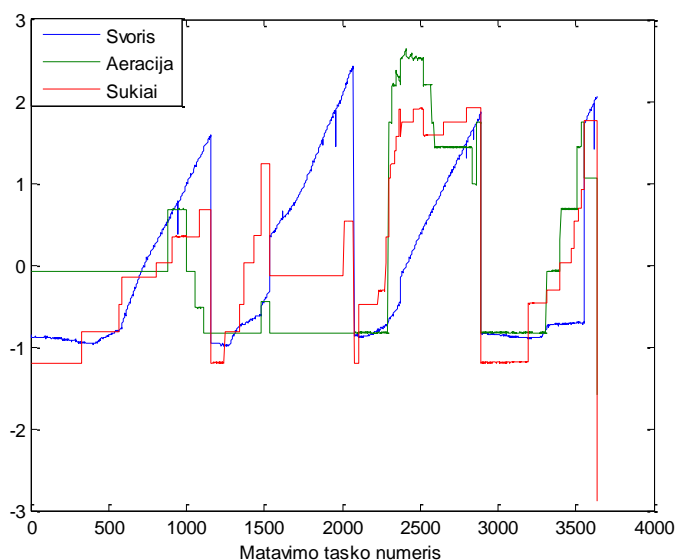
Dirbtinio neuronų tinklo įėjimo vektoriaus ir treniravimo duomenys normuoti pagal nulinį vidurkį ir vienetinę dispersiją, naudojant formulę 2.30

$$M = \frac{1}{N} \sum_{i=1}^N A_i; \quad (2.28)$$

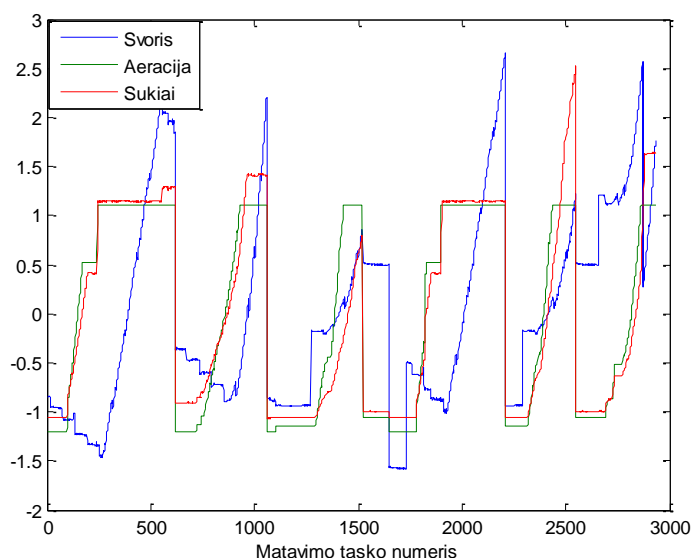
$$S = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |A_i - M|^2}; \quad (2.29)$$

$$y_{\text{norm}} = \frac{y_i - M}{S}; \quad (2.30)$$

Normuoti įėjimo vektoriaus duomenys pavaizduoti 2.8 pav. ir 2.9 pav. Neuronų tinklas turi vienintelį išėjimą, tai $k_L a$ reikšmę.



2.8 pav. Normuoti DNT įėjimo kintamieji, I proceso eksperimentiniai duomenys



2.9 pav. Normuoti DNT įėjimo kintamieji, II proceso eksperimentiniai duomenys

Tyrimo metu mokymui, dėl greito konvergavimo ir stabilumo, pasirinktas Levenberg-Marquardt algoritmas, Matlab ir NNSYSID paketas bei standartinis Matlab neuronų tinklas NNTOOL paketas. Manipuliuojant neuronų skaičiumi paslėptame sluoksnyje, buvo bandoma nustatyti tą neuronų skaičių, ties kuriuo gaunama mažiausia kvadratinė paklaida.

Stacionarios būsenos metodu gauti $k_L a$ duomenys, kurie naudojami neuronų tinklo mokymui ir testavimui, suskirstyti tokia tvarka: visi I proceso 3669 ir II proceso 2934 eksperimentiniai duomenys išdalinti intervalais po 200 matavimo taškų. Kiekviename intervale 75% matavimo taškų skirta mokymui, 25% - testavimui (pavyzdžiui, mokymui [1:150,...], testavimui [151:200,...]). Paslėptame sluoksnyje neuronų keitimo amplitudė svyravo nuo 2 iki 30. Kiekvienam keitimui atlikta po 10 eksperimentų, siekiant nustatyti geriausią prognozavimo rezultatą. Gautų mažiausių kvadratinių paklaidų duomenys pateikti priede Nr.2

Mažiausia kvadratinė paklaida apskaičiuota naudojant formulę 2.31

$$y = \frac{1}{N} \sqrt{\sum_{i=1}^N (y_{ei} - y_{ai})^2}; \quad (2.31)$$

y_{ei} – stacionarios būsenos metodu apskaičiuotos $k_L a$ reikšmės

y_{ai} – neuronų tinklo mokymo metu gautos $k_L a$ reikšmės

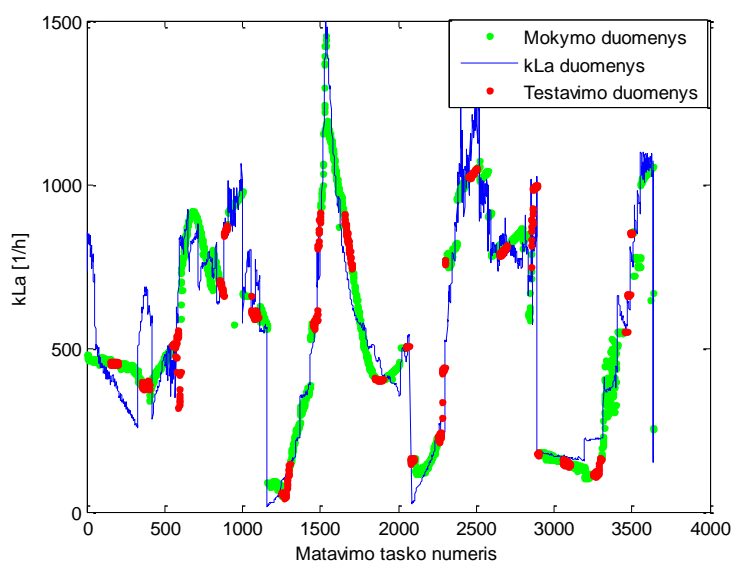
N – matavimo taškų skaičius

3. TYRIMO REZULTATAI

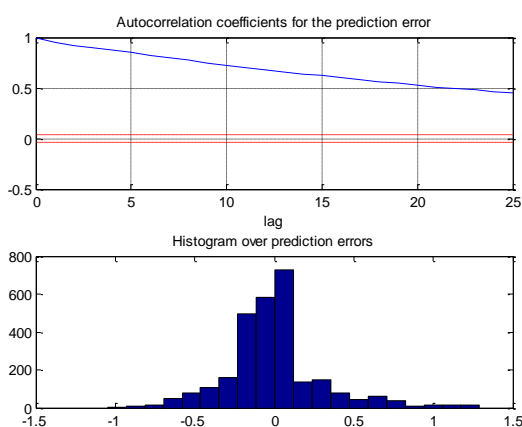
Šio tyrimo metu, naudojant dirbtinius neuronų tinklus, buvo įvertinta $k_L a$ priklausomybė nuo šio parametro vertei įtaką turinčių proceso parametrų, tokių kaip aeracija, maišymas ir kultivavimo terpės tūris. Toliau pateikti rezultatai, rodantys $k_L a$ prognozavimo kokybę priklausomai nuo neuronų skaičiaus paslėptame sluoksnyje, naudojant skirtingų realių procesų eksperimentinius duomenis.

3.1 $k_L a$ reikšmių mokymas ir testavimas naudojant Matlab ir NNSYSID, Levenberg-Marquardt algoritmą

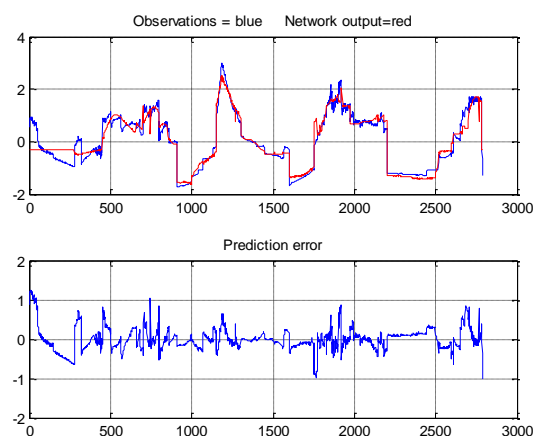
3.1 pav. grafiškai atvaizduoti I proceso DNT $k_L a$ mokymo ir testavimo duomenų imčių prognozavimo rezultatai, gauti naudojant 5 neuronus paslėptame sluoksnyje.



3.1 pav. I proceso $k_L a$ reikšmių modeliavimo rezultatai



3.2 pav. Paklaidos histogramos grafikas

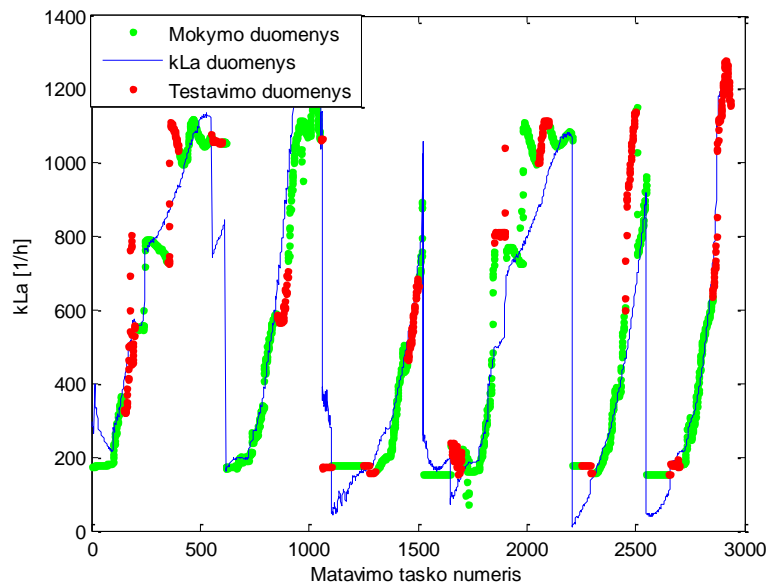


3.3 pav. DNT prognozavimo paklaidos grafikas

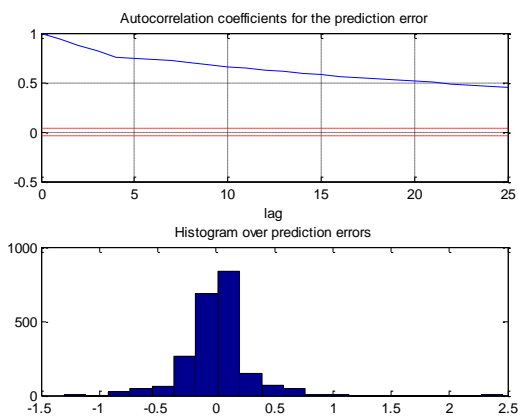
3.1 pav. galima pastebėti mokymo ir testavimo duomenų prognozavimo netikslumus lyginant su realiomis proceso $k_L a$ reikšmėmis. Histogramos grafike 3.2 pav. matome

histogramos kraštus, santykinai plačiai nutolusius nuo nulio, ir pati histogramos forma nėra artima normaliajam skirstiniui. Prognozavimo paklaida išsidėsčiusi simetriškai 0 lygmenis atžvilgiu.

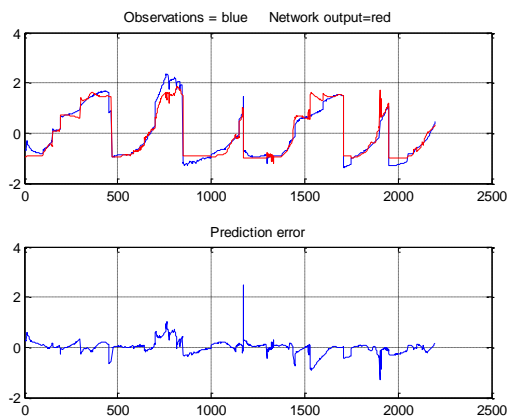
3.4 pav. pateikti grafiškai atvaizduoti II proceso DNT k_La mokymo ir testavimo duomenų imčių prognozavimo rezultatai, gauti naudojant 5 neuronus paslėptame sluoksnyje.



3.4 pav. II proceso k_La reikšmių modeliavimo rezultatai



3.5 pav. Paklaidos histogramos grafikas



3.6 pav. DNT prognozavimo paklaidos grafikas

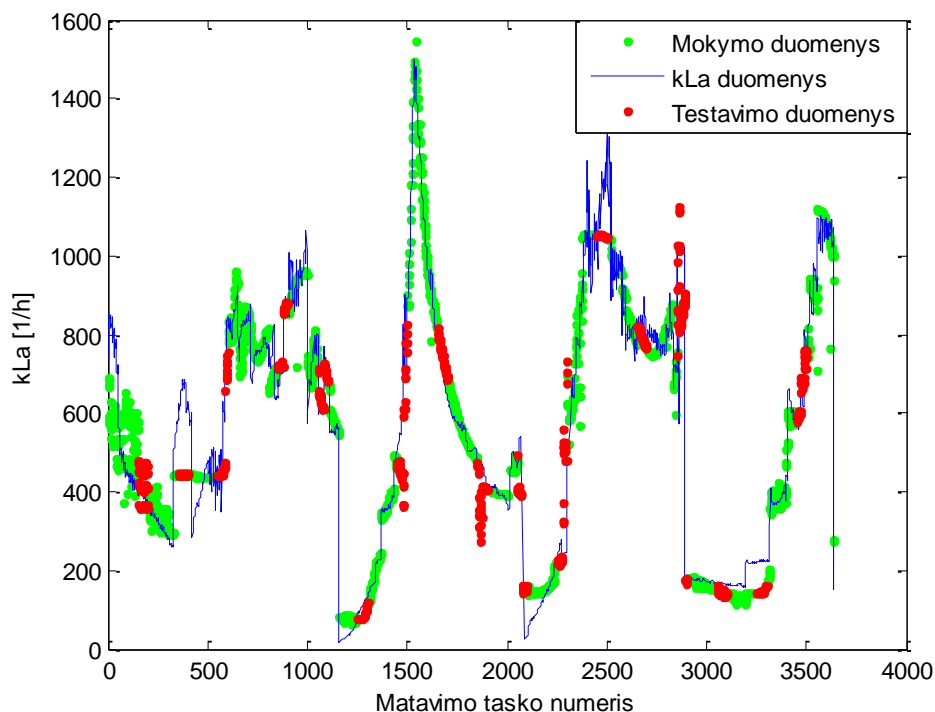
3.4 pav. taip pat pastebimi mokymo ir testavimo duomenų prognozavimo netikslumai, palyginti su realiomis proceso k_La reikšmėmis. Histogramos grafike 3.5 pav. matoma, jog histogramos kraštai nutolę nuo nulio ribose $-1 +1$, histogramos forma nėra artima normaliajam skirstiniui. 3.1 lentelėje pateiktos vidutinės kvadratinės paklaidos, gautos naudojant DNT su 5 neuronais paslėptame sluoksnyje. I proceso metu gauta testavimo duomenų 6,78% paklaida, II proceso 12,44% ir atitinkamai mokymo duomenų paklaida 5,69% ir 7,52%. Šios vertės leidžia sėkmingai taikyti DNT modelius modeliuojamo dydžio (k_La) prognozavimui,

tačiau toliau būtina ištirti DNT struktūros įtaką prognozavimo kokybei ir, jei įmanoma, pagerinti modeliavimo kokybę parenkant optimalios struktūros DNT. Tokių tyrimų rezultatai pateikiami toliau.

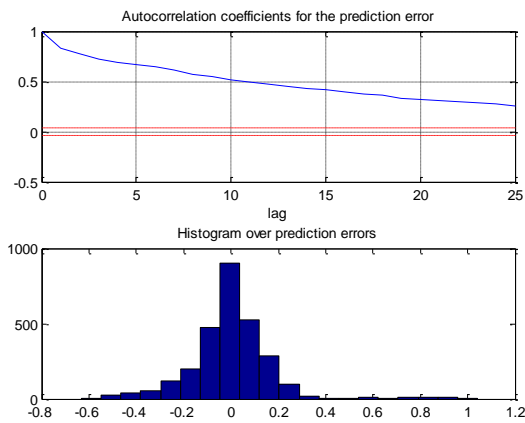
3.1 lentelė

Vidutinė kvadratinė paklaida	I proceso k_La [1/h]	I proceso k_La [%]	II proceso k_La [1/h]	I proceso k_La [%]
Mokymo duomenys	99,91269	5,69	104,7819	7,52
Testavimo duomenys	107,7007	6,78	173,247	12,44

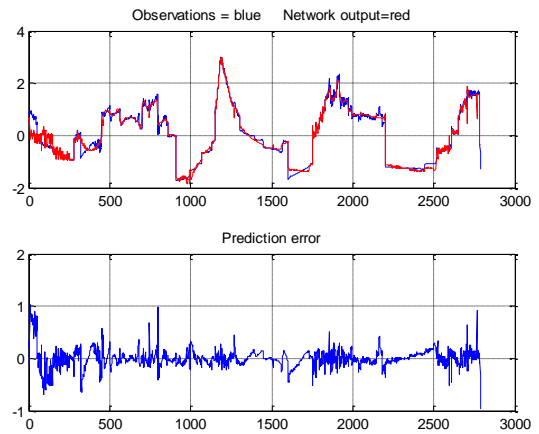
Atlikus įvairios struktūros (su skirtingu neuronų skaičiumi paslėptam sluoksnyje) DNT mokymą ir testavimą, buvo nustatytas optimalios struktūros (minimalios testavimo paklaidos požiūriu) DNT. 3.7 pav. pateikti grafiškai atvaizduoti optimalios struktūros I proceso DNT k_La mokymo ir testavimo duomenų imčių prognozavimo rezultatai, gauti naudojant DNT su 10 neuronų paslėptame sluoksnyje. Nustatyta, kad tokios struktūros DNT užtikrina minimalią I proceso duomenų testavimo paklaidą. Įvairios struktūros DNT modeliavimo paklaidos rezultatai apibendrinti 3.13, 3.15 pav. ir 4 lentelėje.



3.7 pav. k_La modeliavimo rezultatai



3.8 pav. Paklaidos histogramos grafikas



3.9 pav. DNT prognozavimo paklaidos grafikas

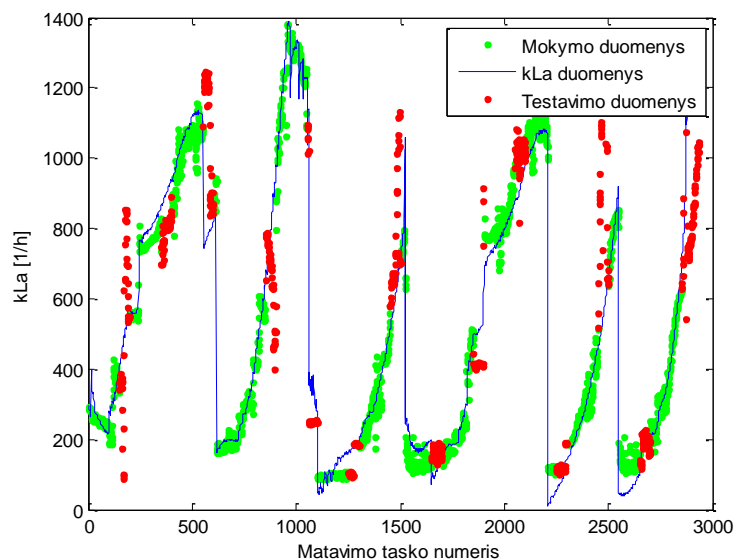
3.7 pav. galima pastebėti mokymo ir testavimo duomenų prognozavimo rezultatų pagerėjimą, lyginant su 3.1 pav. pavaizduotais rezultatais. Histogramos grafike 3.8 pav. matome paklaidos histogramą, kurios ribos yra mažiau nutolusios nuo nulio, ribose -0,6 +0,4, o histogramos forma artimesnė normaliajam skirstiniui. 3.9 pav. pavaizduotos paklaidos vertės taip pat mažesnės, lyginant su 3.3 pav. pateiktas rezultatais.

3.2 lentelėje pateiktos vidutinės kvadratinės paklaidos, gautos naudojant DNT su 10 neuronų paslėptame sluoksnyje (optimali struktūra minimalios testavimo paklaidos požiūriu). I proceso metu gauta testavimo duomenų 5,45% paklaida, mokymo duomenų paklaida 3,75%. Tai geresni rezultatai, palyginti su 5 neuronų paslėptame sluoksnyje DNT prognozavimo rezultatais.

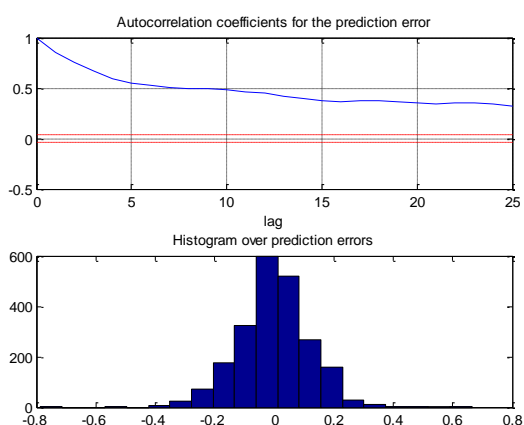
3.2 lentelė

Vidutinė kvadratinė paklaida	I proceso $k_L a$ [1/h]	I proceso $k_L a$ [%]
Mokymo duomenys	56,15381	3,75
Testavimo duomenys	81,75652	5,45

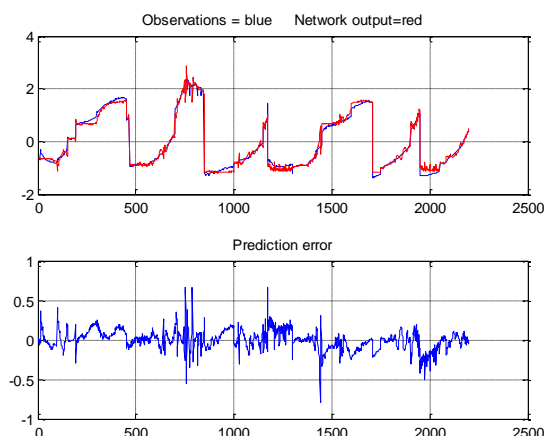
3.10 pav. pateikti grafiškai atvaizduoti optimalios struktūros II proceso DNT $k_L a$ mokymo ir testavimo duomenų imčių prognozavimo rezultatai, naudojant 15 neuronų paslėptame sluoksnyje.



3.10 pav. $k_L a$ modeliavimo rezultatai



3.11 pav. Paklaidos histogramos grafikas



3.12 pav. DNT prognozavimo paklaidos grafikas

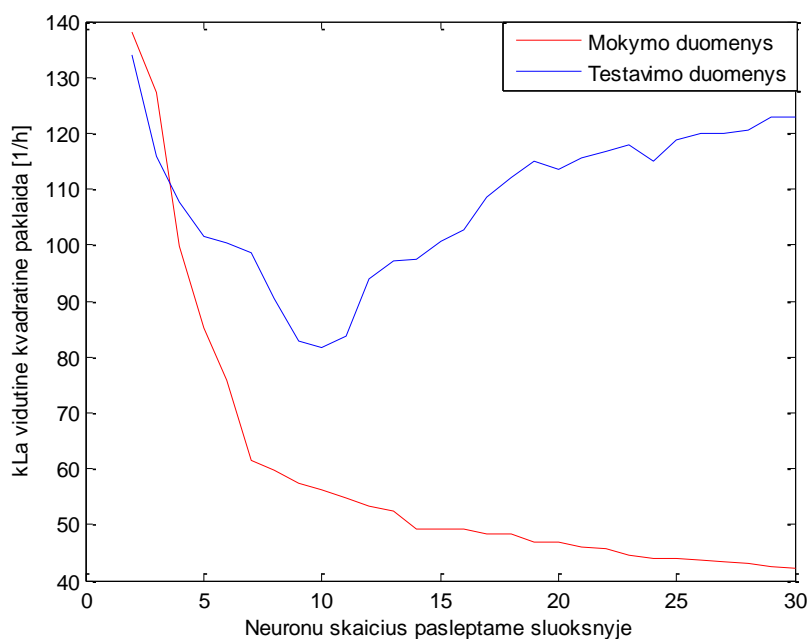
3.10 pav. galima pastebėti mokymo ir testavimo duomenų prognozavimo pagerėjimą, lyginant su apskaičiuotomis $k_L a$ reikšmėmis. 3.11 pav. histogramos grafike histogramos kraštai nutolę mažiau nuo nulio, ribose $-0,4 +0,4$, histogramos forma artima normaliajam skirstiniui.

3.3 lentelėje pateiktos vidutinės kvadratinės paklaidos, gautos naudojant DNT su 15 neuronų paslėptame sluoksnyje. II proceso metu gauta testavimo duomenų 10,93% paklaida, mokymo duomenų paklaida 3,78%.

3.3 lentelė

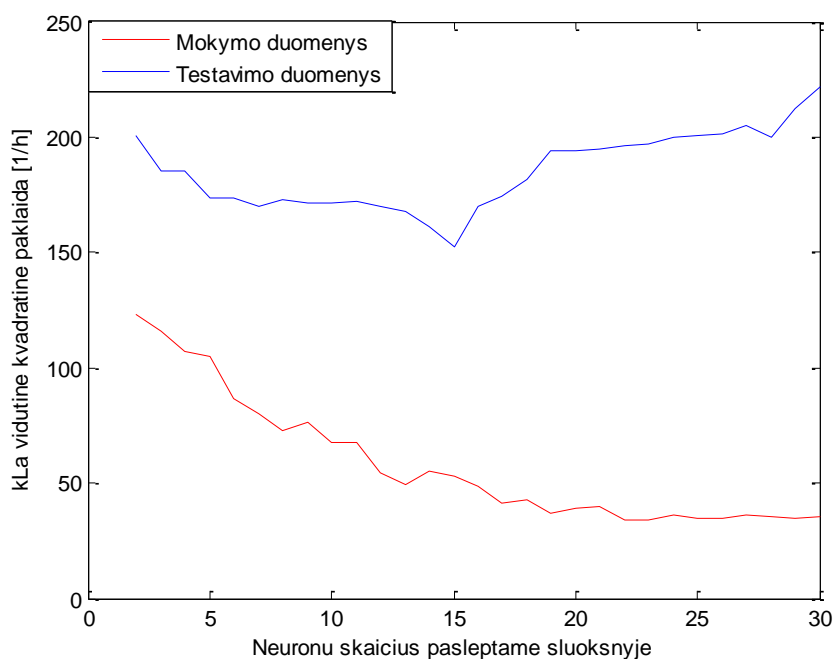
Vidutinė kvadratinė paklaida	II proceso $k_L a$ [1/h]	II proceso $k_L a$ [%]
Mokymo duomenys	52,65771	3,78
Testavimo duomenys	152,1842	10,93

3.13 pav. pateiktas I proceso DNT prognozavimo kokybės rodiklių grafinis atvaizdavimas. Grafike matoma, kokią įtaką prognozavimo kokybei daro neuronų skaičius paslėptame sluoksnyje.



3.13 pav. I proceso vidutinės kvadratinės paklaidos priklausomybės nuo neuronų skaičiaus paslėptame sluoksnyje grafikas

3.14 pateiktas II proceso DNT prognozavimo kokybės rodiklių grafinis atvaizdavimas. Grafike matome, kokią įtaką prognozavimo kokybei daro neuronų skaičius paslėptame sluoksnyje.



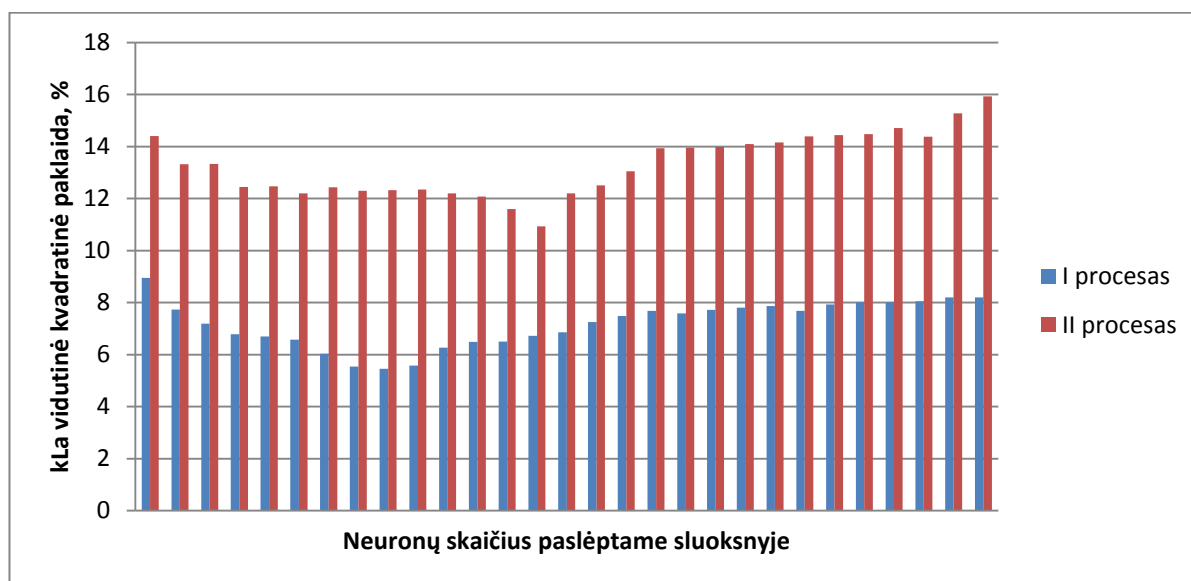
3.14 pav. II proceso vidutinės kvadratinės paklaidos priklausomybės nuo neuronų skaičiaus paslėptame sluoksnyje grafikas

Neuronų tinklo prognozės buvo palygintos su $k_L a$ reikšmėmis, gautomis įprastu duomenų suderinimo metodu. Šie rezultatai pavaizduoti 3.1 – 3.12 pav. $k_L a$ diapazonas dviem fermentacijų sistemoms buvo skirtingas.

Kaip matome iš grafikų 3.13 pav. ir 3.14 pav., didinant neuronų skaičių paslėptame sluoksnyje, mažėja duomenų vidutinė kvadratinė paklaida. I proceso mažiausia kvadratinė paklaida gauta esant 10 neuronų paslėptame sluoksnyje, II proceso – esant 15 neuronų paslėptame sluoksnyje.

Taip pat galima pastebėti, kad didėjant neuronų skaičiui, pradeda didėti testavimo duomenų paklaida, nes, sudėtingėjant tinklui, gerėja aproksimavimo kokybė mokymo duomenyse. Tinklas fiksuoja neesminius sąryšius duomenyse, todėl testavimo duomenyse tinklas ieško neesminių sąryšių.

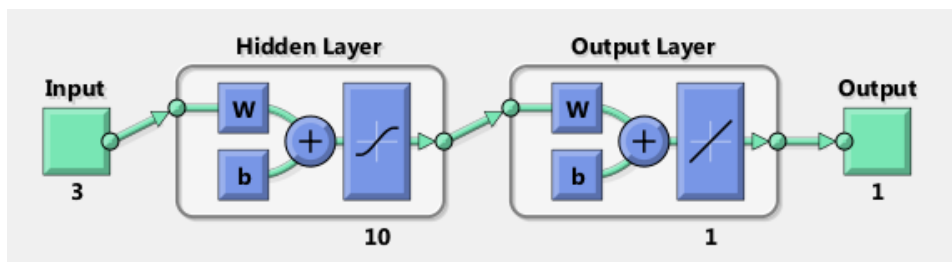
3.15 pav. matoma, kad I procesui yra optimalus 10-ies neuronų skaičius, nes jam buvo gauta mažiausia kvadratinė paklaida – 5,45%, o II procesui – optimalus 15-kos neuronų skaičius, nes jam buvo gauta mažiausia kvadratinė paklaida – 10,93%.



3.15 pav. I ir II procesų vidutinės kvadratinės paklaidos priklausomybės nuo neuronų skaičiaus paslėptame sluoksnyje procentinė išraiška

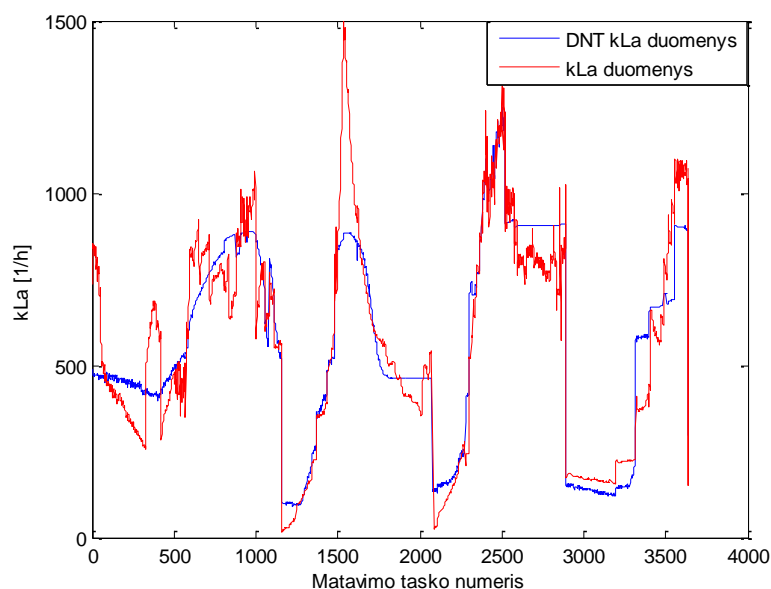
3.2 $k_L a$ prognozavimas per neuronų tinklą, naudojant Matlab ir Neural network tool

Tyrimė taip pat naudojama standartinė Matlab neural network tool dirbtinio neuroninio tinklo versija – daugiasluoksnis tiesioginio sklidimo perceptronas. Neuronų tinklas sudarytas iš 3 sluoksnių. Pirmasis sluoksnis yra jėjimų sluoksnis, turintis tris jėjimus, antrasis sluoksnis yra paslėptas ir trečiasis sluoksnis yra išėjimų sluoksnis, turintis vieną išėjimą $k_L a$ reikšmę.



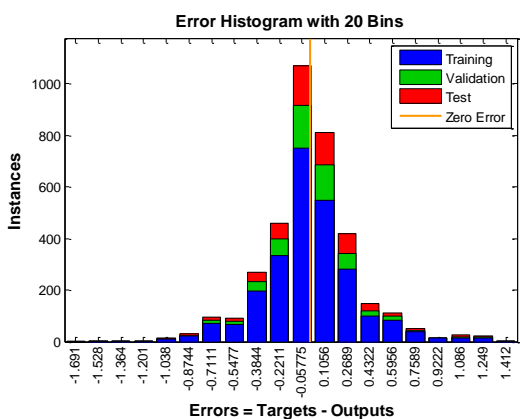
3.16 pav. DNT struktūra

3.17 pav. grafiškai atvaizduotas I proceso DNT $k_L a$ reikšmių prognozavimas naudojant 5 neuronus paslėptame sluoksnyje.

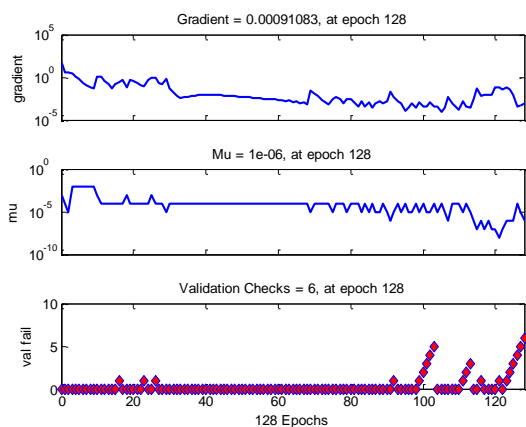


3.17pav. I proceso $k_L a$ modeliavimo rezultatai

3.17 pav. matoma, jog prognozuojamų duomenų aproksimavimas nėra pakankamai geras lyginant su apskaičiuotomis stacionarios būsenos metodu $k_L a$ reikšmėmis. Histogramos grafike 3.18 pav., matoma, jog histogramos kraštai santykinai nutolę nuo centro, ribose $-1,038$ $+1,249$, histogramos forma yra artima normaliajam skirstiniui.

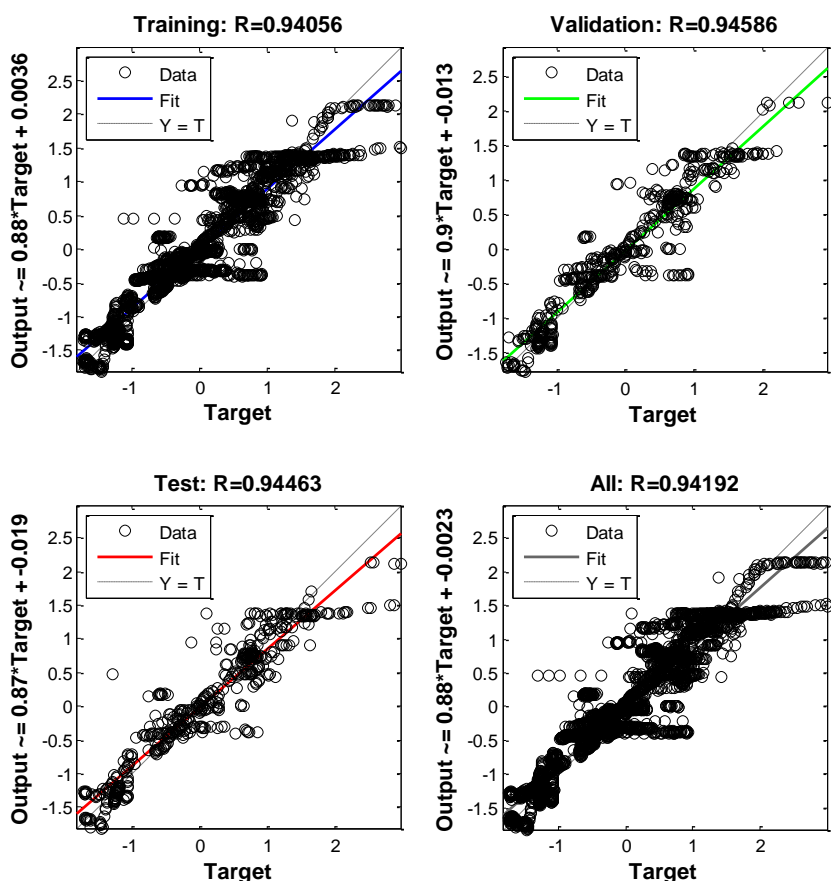


3.18 pav. Paklaidos histograma



3.19 pav. Mokymo būsenos grafikai

Apskaičiuotų $k_L a$ reikšmių ir modeliuotų $k_L a$ reikšmių palyginimo grafike 3.20 pav. matoma, jog mokymo, tikrinimo ir testavimo duomenys nėra santykinai gerai pasiskirstę aplink 45 laipsnių posvyrio tiesę ir yra sisteminių paklaidų.



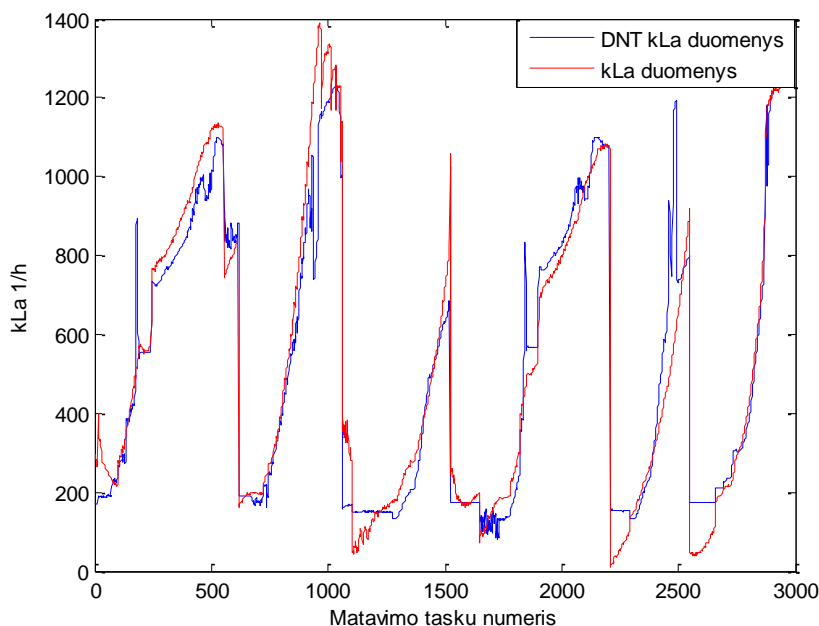
3.20 pav. Apskaičiuotų $k_L a$ reikšmių ir modeliuotų $k_L a$ reikšmių palyginimas

3.4 lentelėje pateiktos vidutinės kvadratinės paklaidos, gautos naudojant DNT su 5 neuronais paslėptame sluoksnyje. I proceso metu gauta testavimo duomenų paklaida - 7,75%.

3.4 lentelė

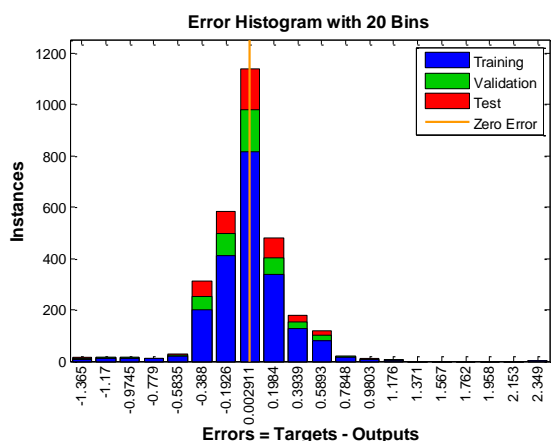
Vidutinė kvadratinė paklaida	I proceso $k_L a$ [1/h]	I proceso $k_L a$ [%]
Testavimo duomenys	116,233	7,75

3.17 pav. grafiškai atvaizduotas II proceso DNT $k_L a$ reikšmių prognozavimas, naudojant 5 neuronus paslėptame sluoksnyje.

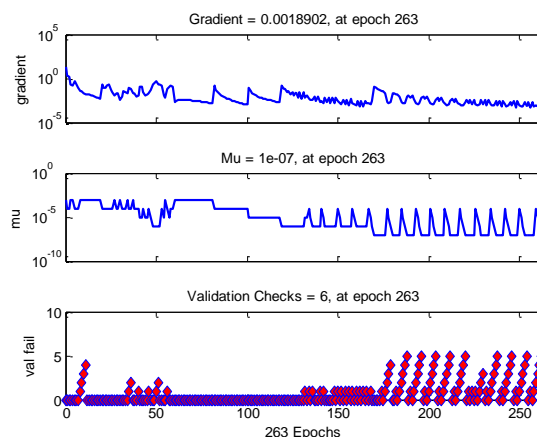


3.21 pav. II proceso $k_L a$ modeliavimo rezultatai .

3.21 pav. matoma, kad prognozuojamų duomenų aproksimavimas nėra labai geras lyginant su apskaičiuotomis stacionarios būsenos metodu $k_L a$ reikšmėmis. Histogramos grafike 3.22 pav. matoma, jog histogramos kraštai yra pakankamai nutolę nuo centro, ribose $-0,583$ $+0,784$, histogramos forma nėra artima normaliajam skirstiniui.

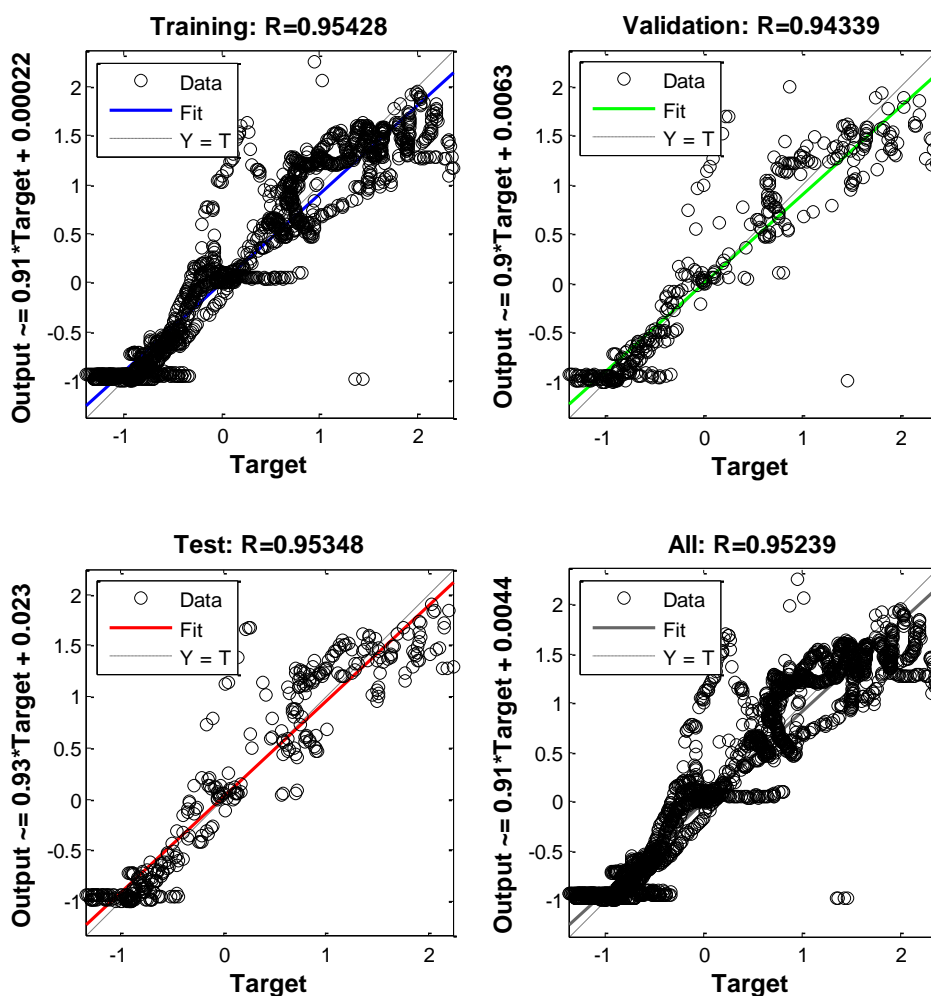


3.22 pav. Paklaidos histograma



3.23 pav. Mokymo būsenos grafikai

Realių proceso $k_L a$ reikšmių ir modeliuotų $k_L a$ reikšmių palyginimo grafike 3.24 pav. matoma, kad mokymo, tikrinimo ir testavimo duomenys nėra simetriškai pasiskirstę aplink 45 laipsnių posvyrio tiesę, pastebima didelė sklaida ir yra sisteminių paklaidų. Nors koreliacijos koeficiento R vertė yra apie 0,95 (pakankamai gera koreliacija), tačiau realių duomenų ir DNT išėjimo tiesinės koreliacijos koeficiento vertė skiriasi mokymo, tikrinimo ir testavimo duomenyse (0,9-0,93) ir ji nėra artima 1.



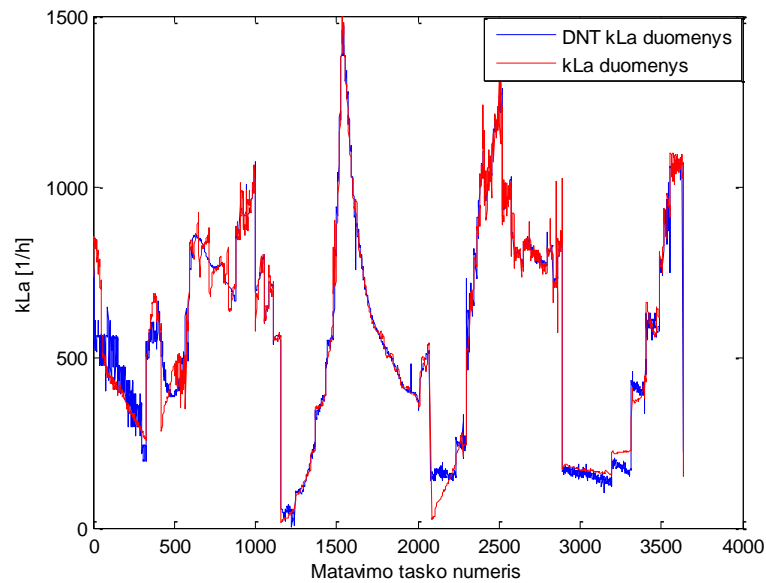
3.24 pav. Apskaičiuotų $k_L a$ reikšmių ir modeliuotų $k_L a$ reikšmių palyginimas

3.5 lentelėje pateiktos vidutinės kvadratinės paklaidos, gautos naudojant DNT su 5 neuronais paslėptame sluoksnyje. II proceso metu gauta testavimo duomenų paklaida lygi 7,38%.

3.5 lentelė

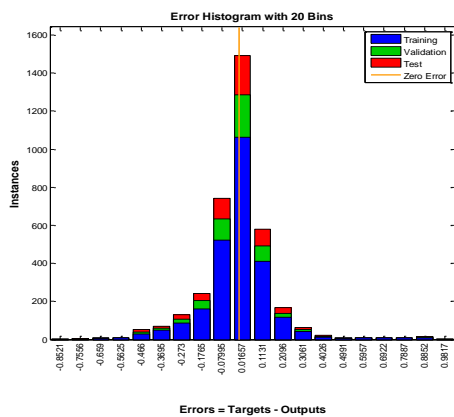
Vidutinė kvadratinė paklaida	II proceso $k_L a$ [1/h]	II proceso $k_L a$ [%]
Testavimo duomenys	102,7693	7,38

3.25 pav. grafiškai atvaizduotas I proceso DNT $k_L a$ reikšmių prognozavimas naudojant 30 neuronų paslėptame sluoksnyje.

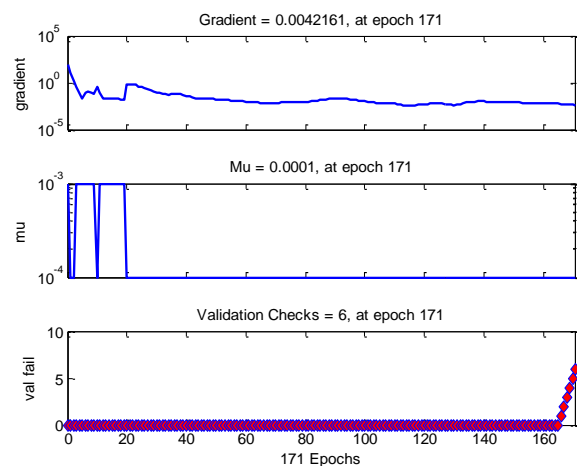


3.25 pav. I proceso $k_L a$ modeliavimo rezultatai

3.25 pav. matomas geras prognozuojamų duomenų aproksimavimas lyginant su stacionarios būsenos metodu apskaičiuotomis $k_L a$ reikšmėmis. Histogramos grafike 3.26 pav. matoma, jog histogramos kraštai yra nutolę nuo centro santykinai siaurose ribose $-0,466 +0,402$, histogramos forma artima normaliajam skirstiniui.



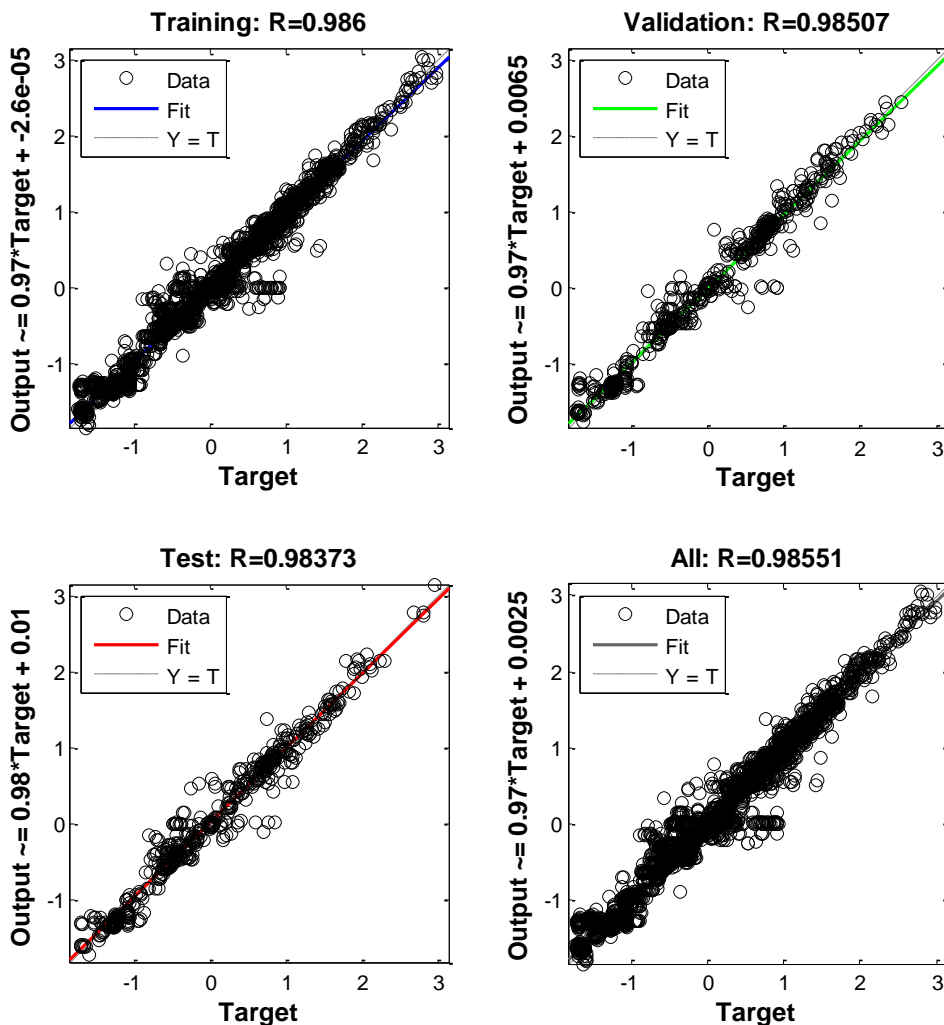
3.26 pav. Paklaidos histograma



3.27 pav. Mokymo būsenos grafikai

Apskaičiuotų $k_L a$ reikšmių ir modeliutų $k_L a$ reikšmių palyginimo grafike 3.28 pav. matoma, kad mokymo, tikrinimo ir testavimo duomenys yra santykinai gerai pasiskirstę aplink

45 laipsnių posvyrio tiesę, sklaida yra ženkliai mažesnė nei 3.24 pav. ir yra mažiau sisteminių paklaidų.



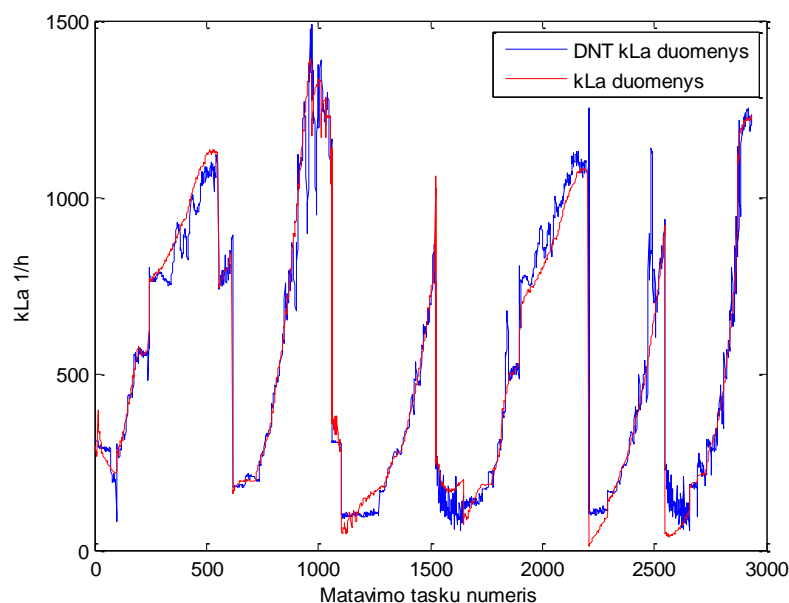
3.28 pav. Apskaičiuotų $k_L a$ reikšmių ir modeliuotų $k_L a$ reikšmių palyginimas

3.5 lentelėje pateiktos vidutinės kvadratinės paklaidos, gautos naudojant DNT su 30 neuronų paslėptame sluoksnyje. I proceso metu gauta testavimo duomenų paklaida lygi 3,41%.

3.5 lentelė

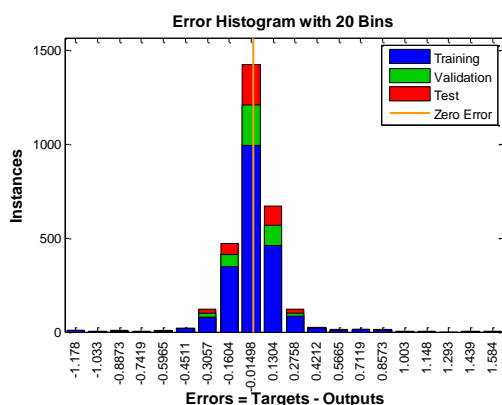
Vidutinė kvadratinė paklaida	I proceso $k_L a$ [1/h]	I proceso $k_L a$ [%]
Testavimo duomenys	51,1679	3,41

3.25 pav. grafiškai atvaizduotas II proceso DNT k_La reikšmių prognozavimas naudojant 30 neuronų paslėpame sluoksnyje.

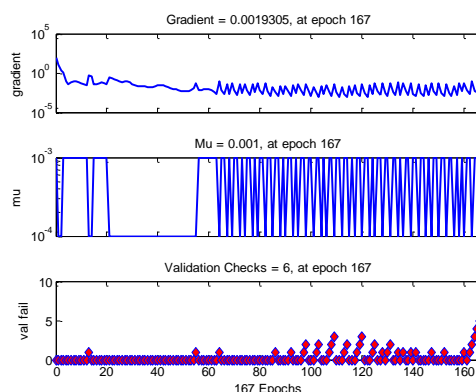


3.29 pav. II proceso k_La modeliavimo rezultatai

3.29 pav. matomas geras prognozuojamų duomenų aproksimavimas lyginant su apskaičiuotomis stacionarios būsenos metodu k_La reikšmėmis. Histogramos grafike 3.30 matoma, kad histogramos kraštai yra nutolę nuo centro ribose $-0,305 +0,0,275$, histogramos forma artima normaliajam skirstiniui.

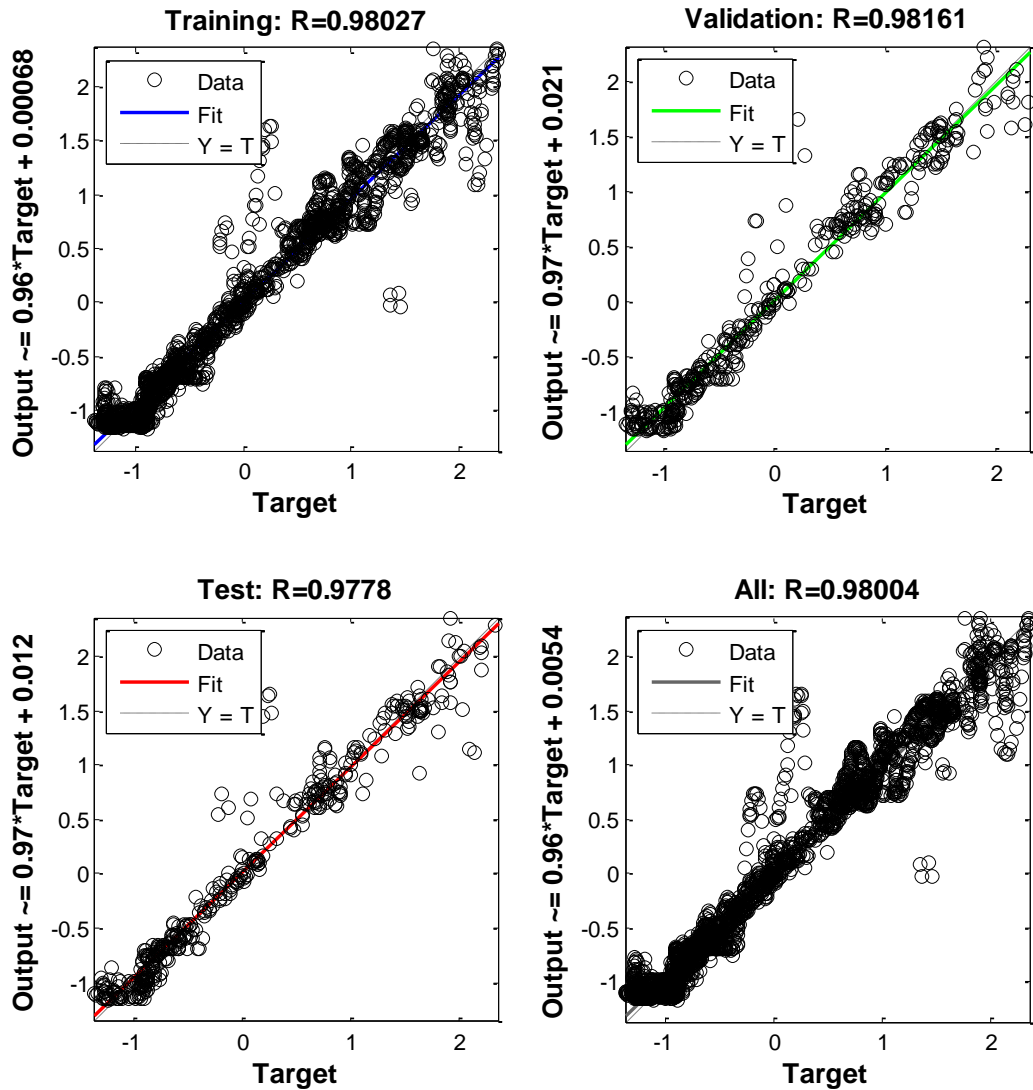


3.30 pav. Paklaidų histograma



3.31 pav. Mokymo būsenos grafikai

Apskaičiuotų k_La reikšmių ir modeliutų k_La reikšmių palyginimo grafike 3.32 pav. matoma, kad mokymo, tikrinimo ir testavimo duomenys yra santykinai gerai pasiskirstę aplink 45 laipsnių tiesę ir mažai akivaizdžių sisteminių paklaidų.



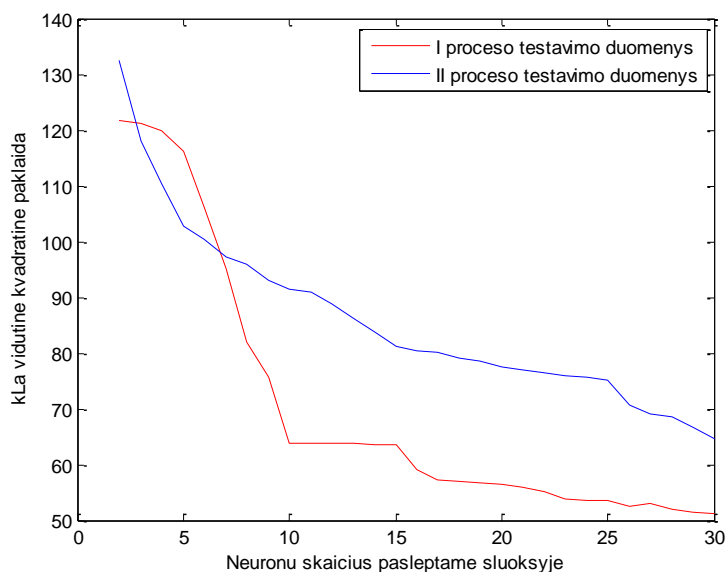
3.32 pav. Apskaičiuotų $k_L a$ reikšmių ir modeliuotų $k_L a$ reikšmių palyginimas

3.6 lentelėje pateiktos vidutinės kvadratinės paklaidos, gautos naudojant DNT su 30 neuronų paslėptame sluoksnyje. II proceso metu gauta testavimo duomenų paklaida lygi 3,41%.

3.6 lentelė

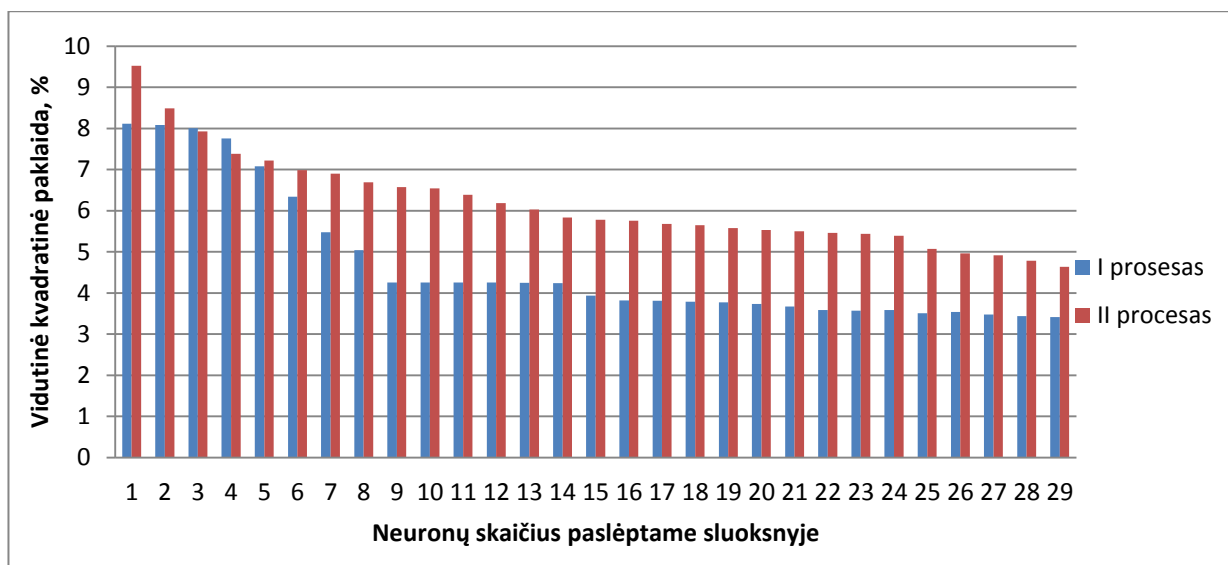
Vidutinė kvadratinė paklaida	II proceso $k_L a$ [1/h]	II proceso $k_L a$ [%]
Testavimo duomenys	64,5729	4,64

Kaip matoma 3.33 pav., eksperimento metu, manipuliuojant neuronų skaičiumi paslėptame sluoksnyje, gaunamas vidutinės kvadratinės paklaidos mažėjimas tiek su I proceso, tiek su II proceso duomenimis.



3.33 pav. Vidutinės kvadratinės paklaidos priklausomybės nuo neuronų paslėptame sluoksnyje skaičiaus grafikas

DNT tinklas gerai prognozuoja $k_L a$ reikšmes didinant neuronų skaičių paslėptame sluoksnyje. Tačiau standartiniame Matlab NNTOOL pakete nepavyko sužinoti, koku būdu yra parenkamas mokymo, tikrinimo, testavimo duomenų intervalų suskirstymas. Yra tikimybė, kad duomenys parenkami atsitiktiniais intervalais, tačiau apmokomi, patikrinami ir testuojame tie, kurie yra vienas arti kito, todėl gaunami tokie geri $k_L a$ prognozavimo rezultatai. Nežinant duomenų suskirstymo intervalų, sunku būtų modelį taikyti realiame procese. 3.33 pav. matoma, kad I ir II procesui geriausi duomenys gauti esant 30-čiai neuronų, kadangi I procesui mažiausia kvadratinė paklaida - 3,41%, o II procesui – 4,64%.



3.34 pav. I ir II procesų vidutinės kvadratinės paklaidos priklausomybės nuo neuronų skaičiaus paslėptame sluoksnyje procentinė išraiška

IŠVADOS

Šiame tyrime atliktas $k_L a$ reikšmės modeliavimas bioreaktorių sistemoje naudojant dirbtinį neuronų tinklą, kuris mokyta duomenimis, nustatytais panaudojant deguonies masės balanso lygtį. Nustatyta, kad neuronų tinklas turi gebėjimą kokybiškai apskaičiuoti $k_L a$ reikšmę iš proceso ir valdymo kintamųjų. Gautos reikšmės buvo artimos reikšmėms, gautoms įprastu, plačiai taikomu, duomenų suderinimo metodu. Naudojant NNSYSID I procesui gauta – 5,45%, o II procesui – 10,93% dydžio paklaidos testavimo imtyse. Naudojant NNTOOL I procesui gauta 3,41%, o II procesui – 4,64% dydžio paklaidos.

Atlikus $k_L a$ prognozavimą per neuronų tinklą, naudojant Matlab ir NNSYSID 2.0 paketą, buvo nustatyta, kad optimalus modelis (atitinkamai I ir II procesams) – turintis dešimt ir penkiolika paslėptų sluoksnių, kadangi šių struktūrų modeliams buvo gauta mažiausia vidutinė kvadratinė patikrinimo paklaida. Toliau didinant neuronų skaičių paslėptame sluoksnyje, vidutinė kvadratinė patikrinimo paklaida pradėjo augti, nes sudėtingėjant tinklui, gerėjo aproksimavimo kokybė apmokymo duomenyse, tačiau tinklas fiksavo neesminius sąryšius duomenyse, todėl patikrinimo duomenyse tinklas ieškojo neesminių sąryšių.

Atlikus $k_L a$ prognozavimą per neuronų tinklą, naudojant NNTOOL paketą, buvo nustatyta, kad didinant neuronų skaičių paslėptame sluoksnyje, tolygiai mažėja vidutinė kvadratinė paklaida. Geriausi rezultatai I ir II procese gauti esant 30-čiai neuronų, tačiau optimalaus neuronų skaičiaus paslėptame sluoksnyje nustatyti nepavyko dėl informacijos apie duomenų suskirstymą NNTOOL programoje trūkumo.

Gauti rezultatai aiškiai įrodė, kad dirbtinių neuronų tinklas gali būti taikomas $k_L a$ reikšmės įvertinimui skirtingoms fermentacijos sistemoms. Sukurti modeliai toliau gali būti naudojami atliekant modeliu pagrįstą bioproceso optimizavimą, kai masės perdavimo procesų intensyvumas yra vienas iš proceso našumą ribojančių veiksnių.

Literatūra:

1. Vytautas Galvanauskas, Donatas Levišauskas „Biotechnologinių procesų modeliavimas, optimizavimas ir valdymas“, 2008, p.129.
2. Vytautas Galvanauskas, Donatas Levišauskas „Biotechnologinių procesų modeliavimas, optimizavimas ir valdymas“ praktikumas ir uždaviniai, 2008, p.134.
3. Magnus Nørgaard, Ole Ravn, Niels K. Poulsen and Lars K. Hansen Nørgaard, Ole Ravn, Niels K. Poulsen and Lars K. Hansen, „Neural Networks for Modelling and Control of Dynamic Systems“ „Springer-Verlag, London, 2000. [žiūrėta 2014m. lapkričio 8d.]. Prieiga per internetą: <http://www.iau.dtu.dk/nnspringer.html>.
4. MATLAB Neural network toolbox. [žiūrėta 2014m. lapkričio 8d.]. Prieiga per internetą: <http://se.mathworks.com/products/neural-network/>.
5. Najafpour D.G. „Biochemical engineering and biotechnology“, 2007m. p. 421.
6. Doran M. „Bioprocess engineering principles“, 2013m. p. 919.
7. David A. Mitchell, Oscar F. Von Meien, Nadia Krieger „Recent developments in modeling of solid-state fermentation: heat and mass transfer in bioreactors“, 2002m. [žiūrėta 2014m. gegužės 1d.]. Prieiga per internetą: <http://www.sciencedirect.com/science/article/pii/S1369703X02001262>
8. A.C. Badino Jr., M.C.R. Racciotti, W. Schmidell “Volumetric oxygen transfer coefficients k_La in batch cultivation involving non-Newtonian broth”, 2001m. [žiūrėta 2014m. gegužės 1d.]. Prieiga per internetą: <http://www.sciencedirect.com/science/article/pii/S1369703X01000924>
9. Pisut Tainmanakul, Jidapa Wachirasak, Marupatch Jamnongwong, Giles Hebrard “Theoretical prediction of volumetric mass transfer coefficient k_La for designing an aeration tank”, 2009m. [žiūrėta 2014m. gegužės 1d.]. Prieiga per internetą: <http://www.engj.org/index.php/ej/article/view/64/31>
10. Nilesh Patel, Jules Thibault „Data reconciliation using neural networks for the determination of k_La “, 2009m. [žiūrėta 2014m. gegužės 1d.]. Prieiga per internetą: http://link.springer.com/chapter/10.1007%2F978-3-642-01888-6_7.
11. Bishop, Ch. M. Neural Networks for Pattern Recognition. – Oxford: Clarendon press, 1997. p. 482.
12. Looney, C. G. Pattern recognition using neural networks: theory and algorithms for engineers and scientists. – New York: Oxford University Press, 1997. p. 458.

13. Nurashikin Suhaili, Mohd Shamzi Mohomed, Rosfarizan Mohamad, Arbakariya B. Ariff „ Gas-liquid Transfer Performance of dual Impeller System Employing Rushton’s, concave-blade disc turbines and their combination in stirred tank bioreactor“ 2010. [žiūrēta 2014m. gegužēs 1d.]. Prieiga per internetu: <http://psasir.upm.edu.my/14846/>.
14. Ali Karimi, Farideh Golbabaie, Momammad Reza Mehrnia, Masoud Neghab, Kazem Mohamad, Ahmad Nikpey, Mohammad Reza Pourmand „, Oxygen mass transfer in a stirred tank bioreactor using different impeller configurations for environmental purposes“ 2013. [žiūrēta 2014m. gegužēs 1d.]. Prieiga per internetu: <http://www.ijehse.com/content/10/1/6>
15. D. Moutafchieva, D. Popova, M. Dimitrova, S. Tchaoushev „, Experimental determination of the volumetric mass transfer coefficient“ 2013. [žiūrēta 2014m. gegužēs 1d.]. Prieiga per internetu: <http://dl.uctm.edu/journal/node/j2013-4/4-Chaushev%20-%20351-356.pdf>.
16. S. Schaepe, A. Kuprijanov, C. Sieblist, M. Jensch, R. Simutis, A. Lubbert „, kLa of stirred tank bioreactors revisited“ 2013. [žiūrēta 2014m. gegužēs 1d.]. Prieiga per internetu: <http://www.sciencedirect.com/science/article/pii/S0168165613003830>

Priedai

Priedas Nr.1. I proceso DNT tinklo mokymo ir testavimo duomenų vidutinės kvadratinės paklaidos naudojant Matlab ir NNSYSID

4.1 lentelė

Neuronų skaičius paslėptame sluoksnyje	Vidutinė kvadratinė paklaida			
	Mokymo duomenys [1/h]	Mokymo duomenys [%]	Testavimo duomenys [1/h]	Testavimo duomenys [%]
2	138,242	9,23	134,015	8,95
3	127,356	8,50	115,785	7,73
4	99,9127	6,67	107,701	7,19
5	85,0832	5,68	101,581	6,78
6	75,7587	5,06	100,332	6,70
7	61,4324	4,10	98,5403	6,58
8	59,7704	3,99	90,3395	6,03
9	57,2779	3,82	82,9285	5,54
10	56,1538	3,75	81,7565	5,46
11	54,8618	3,66	83,5723	5,58
12	53,337	3,56	93,8219	6,26
13	52,306	3,49	97,1617	6,49
14	49,3392	3,29	97,3118	6,50
15	49,3039	3,29	100,646	6,72
16	49,307	3,29	102,725	6,86
17	48,4561	3,23	108,624	7,25
18	48,3278	3,23	112,2	7,49
19	46,9679	3,14	115,065	7,68
20	46,9222	3,13	113,567	7,58
21	46,1267	3,08	115,663	7,72
22	45,6289	3,05	116,891	7,80
23	44,4433	2,97	117,885	7,87
24	43,9923	2,94	115,082	7,68
25	43,921	2,93	118,776	7,93
26	43,645	2,91	120,071	8,01
27	43,4409	2,90	120,091	8,02
28	43,0044	2,87	120,56	8,05
29	42,5998	2,84	122,772	8,20
30	42,1648	2,81	122,784	8,20

Priedas Nr.2. II proceso DNT tinklo mokymo ir testavimo duomenų vidutinės kvadratinės paklaidos naudojant Matlab ir NNSYSID

4.2 lentelė

Neuronų skaičius paslėptame sluoksnyje	Vidutinė kvadratinė paklaida			
	Mokymo duomenys [1/h]	Mokymo duomenys [%]	Testavimo duomenys [1/h]	Testavimo duomenys [%]
2	122,89	8,83	200,442	14,40
3	115,544	8,30	185,268	13,31
4	107,215	7,70	185,483	13,33
5	104,782	7,53	173,247	12,45
6	86,8099	6,24	173,473	12,47
7	79,7225	5,73	169,824	12,20
8	72,5699	5,21	172,944	12,43
9	76,33	5,49	171,075	12,29
10	67,9523	4,88	171,508	12,32
11	67,7683	4,87	171,785	12,34
12	54,4221	3,91	169,826	12,20
13	49,1572	3,53	168,106	12,08
14	55,5317	3,99	161,39	11,60
15	52,6577	3,78	152,184	10,94
16	48,7042	3,50	169,756	12,20
17	41,5204	2,98	174,079	12,51
18	42,6767	3,07	181,59	13,05
19	36,611	2,63	193,928	13,94
20	39,2078	2,82	194,236	13,96
21	39,8565	2,86	194,571	13,98
22	34,3098	2,47	196,119	14,09
23	34,2018	2,46	197,003	14,16
24	36,2368	2,60	200,221	14,39
25	34,9182	2,51	200,974	14,44
26	34,7516	2,50	201,42	14,47
27	36,2734	2,61	204,705	14,71
28	35,4612	2,55	200,007	14,37
29	34,8652	2,51	212,585	15,28
30	35,1738	2,53	221,589	15,92

Priedas Nr.3. I ir II proceso DNT tinklo testavimo duomenų vidutinės kvadratinės paklaidos naudojant Matlab ir Nntool

4.3 lentelė

Neuronų skaičius paslėptame sluoksnyje	Vidutinė kvadratinė paklaida			
	I Proceso testavimo duomenys [1/h]	I Proceso testavimo duomenys [%]	II proceso testavimo duomenys [1/h]	II proceso testavimo duomenys [%]
2	121,6044	8,12	132,489	9,52
3	121,1089	8,08	118,1295	8,49
4	119,875	8,00	110,2736	7,92
5	116,233	7,76	102,7693	7,38
6	106,117	7,08	100,4762	7,22
7	95,004	6,34	97,2132	6,99
8	81,994	5,47	95,9983	6,90
9	75,529	5,04	93,1504	6,69
10	63,7951	4,26	91,4996	6,58
11	63,7789	4,26	91,0003	6,54
12	63,7102	4,25	88,9001	6,39
13	63,6999	4,25	86,1078	6,19
14	63,6276	4,25	83,8899	6,03
15	63,5636	4,24	81,2245	5,84
16	59,0272	3,94	80,3894	5,78
17	57,2693	3,82	80,1663	5,76
18	57,1032	3,81	79,0387	5,68
19	56,7566	3,79	78,6293	5,65
20	56,541	3,77	77,5866	5,58
21	55,9673	3,74	77,0048	5,53
22	55,0056	3,67	76,4895	5,50
23	53,7221	3,59	76,0086	5,46
24	53,4433	3,57	75,6657	5,44
25	53,6711	3,58	75,0150	5,39
26	52,6001	3,51	70,5379	5,07
27	52,9999	3,54	69,0227	4,96
28	52,0431	3,47	68,4598	4,92
29	51,5476	3,44	66,5827	4,78
30	51,1679	3,42	64,5729	4,64

Priedas Nr.4. Matlab programa

```
%%%%%I ir II Proceso duomenų filtravimas%%%%%
clear;
clc;
close all;
Cpr1=7.52e-03; %Oxygen solubility
% Load data from fileload 'recl.dat';
our_OTR=recl(:,5);
weight=recl(:,20);
pO2=recl(:,7);
airflow=recl(:,11);
CO2=recl(:,2);
O2=recl(:,3);
stirr=recl(:,13);
kLa_duom=recl(:,8)

F=[];

m1=length(recl(:,1));
for n1=1:m1;
    t2(n1,1)=n1;
    our2(n1,1)=our_OTR(n1,1);
    C2(n1,1)=(pO2(n1,1)/100)*Cpr1;
    kLa2(n1,1)=our2(n1,1)/(Cpr1-C2(n1,1));
    Cp(n1,1)=Cpr1;

    if kLa2(n1,1) < 2000
        F=[F;recl(n1,:)];
        kLa1(n1,1) = 0;
    end
end
M=max(pO2)
G=min(pO2)
save('reclfl.dat','F','-ascii');

%%%%Duomenų normavimas %%%%%%
clear all;
clc;
close all;
Cpr1=7.52e-03; %Oxygen solubility
% Load data from file
load 'duomenys.dat';
our2=duomenys(:,1);
Cp=duomenys(:,2);
C2=duomenys(:,3);
weight=duomenys(:,4);
pO2=duomenys(:,5);
airflow=duomenys(:,6);
CO2=duomenys(:,7);
O2=duomenys(:,8);
stirr=duomenys(:,9);
kLa2=duomenys(:,10);
m1=length(duomenys(:,1))
for n1=1:m1;
    t2(n1,1)=n1;
end
duomen=[ our2 Cp C2 weight pO2 airflow CO2 O2 stirr];
%%Duomenų atvaizdavimas%%
subplot(2,4,1)
plot(t2, our2)
xlabel('our')
```

```

subplot (2,4,2)
plot ( t2, kLa2 )
xlabel ('kLa')
subplot (2,4,3)
plot (t2, airflow)
xlabel ('airflow')
subplot (2,4,4)
plot (t2, CO2)
xlabel ('CO2')
subplot (2,4,5)
plot (t2, O2)
xlabel ('O2')
subplot (2,4,6)
plot (t2, stirr)
xlabel ('stirr')
subplot (2,4,7)
plot (t2, pO2)
xlabel('pO2')
subplot (2,4,8)
plot (t2, weight)
xlabel ('weight')
figure (2)
plot (t2, kLa2)
legend ('kLa duomenys')
xlabel('Matavimo tasko numeris')
ylabel('kLa [1/h]')
figure(3)
plot(t2,weight*110,t2,airflow,t2,stirr)
legend ('weight', 'aiflow', 'stirr')
%%%Duomenų normavimas%%%
klastd=std(kLa2)
klamean=mean(kLa2)
klanorm=(kLa2-klamean)./klastd
save ('kla.dat', 'klanorm', '-ascii')
%%%Vidurkio ir standartinio nuokrupio skaičiavimas%%%
meandata=mean (duomen)
stddata=std (duomen)
%%%Duomenų normavimas%%%
ournorm=(duomen (:,1)-meandata (1))./stddata (1)
cpnormdata=(duomen (:,2)-meandata (2))./stddata (2)
c2normdata=(duomen (:,3)-meandata (3))./stddata (3)
weightnormdata=(duomen (:,4)-meandata (4))./stddata (4)
pO2normdata=(duomen (:,5)-meandata (5))./stddata (5)
airflownormdata=(duomen (:,6)-meandata (6))./stddata (6)
CO2normdata=(duomen (:,7)-meandata (7))./stddata (7)
O2normdata=(duomen (:,8)-meandata (8))./stddata (8)
stirrnormdata=(duomen (:,9)-meandata (9))./stddata (9)
input=[ournorm cpnormdata c2normdata weightnormdata pO2normdata
airflownormdata CO2normdata O2normdata stirrnormdata ]
save ('duomen.dat', 'duomenys', '-ascii');
save ('input.dat', 'input', '-ascii');
save ('klamean.dat', 'klamean', '-ascii');
save ('klastd.dat', 'klastd', '-ascii');
fff=max(kLa2)

%%%DNT NNSYSID2.0%%
clear all, close all, clc

for dd=1:10;
%%%Duomenų uzkrovimas%%
load('input.dat')
load('kla.dat')
weight=input(:,4);

```

```

airflow=input(:,6);
stirr=input(:,9);
kLanormdata=kla(:,1);

%%Mokymo duomeniu suskirstymas%%
itraining=[1:150, 201:350, 401:550, 601:850, 901:1050, 1101:1250,
1301:1450, 1501:1650, 1701:1850, 1901:2050, 2101:2250, 2301:2450, 2501:2650,
2701:2850
%%Treniravimo duomeniu suskirstymas%%
ivalid=[151:200, 351:400, 551:600, 851:900, 1051:1100, 1251:1300,
1451:1500, 1651:1700, 1851:1900, 2051:2100, 2251:2300, 2451:2500, 2651:2700,
2851:2934

%%Iejimo vektorius%%
input_M=[weight(itraining) airflow(itraining) stirr(itraining)'];

%%Isejimo vektorius%%
output_M=[kLanormdata(itraining)'];

%%DNT%%

number_inputs=3;
nodes_hidden_layer=15;
nodes_output_layer=1;
NetDef=['HHHHHHHHHHHHHHHHH-';'L-----'];

W1=rand(nodes_hidden_layer, number_inputs+1)-0.5;
W2=rand(nodes_output_layer, nodes_hidden_layer+1)-0.5;
M_options= [200 0 1 0];

%parametru priskirimas
TRPARMS=settrain(settrain, 'maxiter',5000);
[W1A,W2A,err,it,lambda]=marq(NetDef,W1,W2,input_M,output_M,TRPARMS);

%[W1A,W2A,err,it,lambda]=MARQXP_2(NetDef,W1,W2,input_M,output_M,M_option
s)
load ('klastd.dat');
load ('klamean.dat');
stddata=klastd;
meandata=klamean;

d=length(input(:,1));
for i=1:d
    tt(i,1)=i;
end

%evaliution-training
kLanorm = nneval(NetDef,W1A,W2A,input_M,output_M);
kLaff=kLanorm.*stddata+meandata;
kLamod=[kLaff]';
kLamat=kla.*stddata+meandata;

%evaliution-validation
input_M=[weight(ivalid) airflow(ivalid) stirr(ivalid)'];
output_M=[kLanormdata(ivalid)'];
kLanorm2 = nneval(NetDef,W1A,W2A,input_M,output_M);
kLaff=kLanorm2.*stddata+meandata;
kLamod2=[kLaff]';

%Duomeniu atvaizdavimas%
figure (3)
plot(tt,weight,tt,airflow, tt, stirr)

```

```

legend ('Svoris', 'Aeracija', 'Sukiai')
xlabel('Matavimo tasko numeris')
figure (4)
plot (tt(itraining), kLamod, 'g.', tt, kLamat, 'b', tt(ivalid), kLamod2, 'r.')
legend ('Mokymo duomenys', 'kLa duomenys', 'Testavimo duomenys')
xlabel('Matavimo tasko numeris')
ylabel('kLa [1/h]')

figure (5)
plot (tt, kLamat, 'b')
legend ('kLa duomenys')
xlabel('Laikas')
ylabel('kLa [1/h]')

%%%Vidutines kvadratine paklaidos skaičiavimas%%%
dattrain=(kLamod-kLamat(itraining)).^2;
datatr=sqrt(sum(dattrain)/length(itraining))
datvalid=(kLamod2-kLamat(ivalid)).^2;
datavalid=sqrt(sum(datvalid)/length(ivalid))

simin=weight';
simin1=airflow';
simin2=stirr';

if datavalid < 173;
    stop
end
ner(dd,1)=datatr;

ner(dd,2)=datavalid;

clearvars -except dd ner
end

mintr=min(ner(:,1))
minval=min(ner(:,2))

%Duomenų grafinis atvaizdavimas%
clc, clear all, close all
load ('vid_pakld.txt')
a=vid_pakld(:,1)
b=vid_pakld(:,2)
c=vid_pakld(:,3)
plot(a,b,'r',a,c,'b')
legend('Mokymo duomenys', 'Testavimo duomenys')
xlabel('Neuronų skaičius pasleptame sluoksnyje')
ylabel('kLa vidutine kvadratine paklaida [1/h]')

% Solve an Input-Output Fitting problem with a Neural Network
% Script generated by NFTOOL
% Created Thu May 14 14:10:36 EEST 2015
% This script assumes these variables are defined:
% nntool_duomenys - input data.
% kla - target data.

inputs = nntool_duomenys';
targets = kla';

% Create a Fitting Network
hiddenLayerSize = 10;

```

```

net = fitnet(hiddenLayerSize);

% Setup Division of Data for Training, Validation, Testing
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% Train the Network
[net,tr] = train(net,inputs,targets);

% Test the Network
outputs = net(inputs);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs)

% View the Network
view(net)

% Plots
% Uncomment these lines to enable various plots.
%figure, plotperform(tr)
%figure, plottrainstate(tr)
%figure, plotfit(net,inputs,targets)
%figure, plotregression(targets,outputs)
%figure, ploterrhist(errors)

% Solve an Input-Output Fitting problem with a Neural Network
% Script generated by NFTOOL
% Created Thu May 14 14:11:25 EEST 2015
%
% This script assumes these variables are defined:
%
% nntool_duomenys - input data.
% kla - target data.

inputs = nntool_duomenys';
targets = kla';

% Create a Fitting Network
hiddenLayerSize = 10;
net = fitnet(hiddenLayerSize);

% Choose Input and Output Pre/Post-Processing Functions
% For a list of all processing functions type: help nprocess
net.inputs{1}.processFcns = {'removeconstantrows','mapminmax'};
net.outputs{2}.processFcns = {'removeconstantrows','mapminmax'};

% Setup Division of Data for Training, Validation, Testing
% For a list of all data division functions type: help nndivide
net.divideFcn = 'dividerand'; % Divide data randomly
net.divideMode = 'sample'; % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;

% For help on training function 'trainlm' type: help trainlm
% For a list of all training functions type: help nntain
net.trainFcn = 'trainlm'; % Levenberg-Marquardt

% Choose a Performance Function

```

```

% For a list of all performance functions type: help nnperformance
net.performFcn = 'mse'; % Mean squared error

% Choose Plot Functions
% For a list of all plot functions type: help nnplot
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
    'plotregression', 'plotfit'};

% Train the Network
[net,tr] = train(net,inputs,targets);

% Test the Network
outputs = net(inputs);
errors = gsubtract(targets,outputs);
performance = perform(net,targets,outputs)

% Recalculate Training, Validation and Test Performance
trainTargets = targets .* tr.trainMask{1};
valTargets = targets .* tr.valMask{1};
testTargets = targets .* tr.testMask{1};
trainPerformance = perform(net,trainTargets,outputs)
valPerformance = perform(net,valTargets,outputs)
testPerformance = perform(net,testTargets,outputs)

% View the Network
view(net)

% Plots
% Uncomment these lines to enable various plots.
%figure, plotperform(tr)
%figure, plottrainstate(tr)
%figure, plotfit(net,inputs,targets)
%figure, plotregression(targets,outputs)
%figure, ploterrhist(errors)

```