



KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS

Justinas Bačys

**TIPINIŲ PROCESŲ MODELIŲ IDENTIFIKAVIMO METODŲ
ANALIZĖ IR TYRIMAS**

Baigiamasis magistro projektas

Vadovas

prof. dr. Rimvydas Simutis

KAUNAS, 2017

KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS
AUTOMATIKOS KATEDRA

TIPINIŲ PROCESŲ MODELIŲ IDENTIFIKAVIMO METODŲ
ANALIZĖ IR TYRIMAS

Baigiamasis magistro projektas

Valdymo technologijos (621H66001)

Vadovas

prof. dr. Rimvydas Simutis

Recenzentas

prof. dr. Vytautas Galvanauskas

Projektą atliko

Justinas Bačys

KAUNAS, 2017



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Elektros ir elektronikos fakultetas

(Fakultetas)

Justinas Bačys

(Studento vardas, pavardė)

Valdymo technologijos, 621H66001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Tipinių procesų modelių identifikavimo metodų analizė ir tyrimas“

AKADEMINIO SAŽINGUMO DEKLARACIJA

20 _____ . _____ . _____ .
Kaunas

Patvirtinu, kad mano **Justino Bačio** baigiamasis projektas tema „Tipinių procesų modelių identifikavimo metodų analizė ir 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)

Bačys Justinas. Tipinių procesų modelių identifikavimo metodų analizė ir tyrimas. Valdymo sistemų magistro baigiamasis projektas / vadovas prof. dr. Rimvydas Simutis; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas, Automatikos katedra.

Mokslo kryptis ir sritis: Elektros ir elektronikos inžinerija, Technologiniai mokslai

Reikšminiai žodžiai: identifikavimas, slenkantis duomenų langas, uždaro kontūro reguliavimo sistema.

Kaunas, 2017. 76 p.

SANTRAUKA

Šiame darbe tiriami procesų identifikavimo metodai, kuriuos galima taikyti siekiant identifikuoti tipinius procesus iš normalaus funkcionavimo duomenų.

Pagrindinis darbo tikslas atlikti tipinių procesų modelių identifikavimo iš realaus funkcionavimo duomenų algoritmų analizę ir naudojant identifikuotus parametrus realizuoti proceso kintamųjų reguliavimo sistemas. Procesų identifikavimui pasirinkti du stabilūs identifikavimo metodai ir naudojant visus proceso duomenis ištirta, kokią įtaką proceso parametrų identifikavimui turi duomenų diskretiškumas ir triukšmo lygis.

Toliau atlikti proceso parametrų identifikavimo eksperimentai su slenkančiu duomenų langu ir patikrintas metodų tinkamumas parametrų nustatymui. Kitame projekto etape atlikti eksperimentai su uždaro kontūro reguliavimo sistema. Pasitelkiant proceso parametrų identifikavimo metodą buvo įvertinami proceso modelio parametrai, keičiami reguliatoriaus parametrai realiu laiku ir lyginama reguliavimo kokybė su sistemas, kurioje naudojami pastovūs reguliatoriaus parametrai kokybe.

Atlikti eksperimentai parodė, kad procesų reguliavimo kokybė gali būti pagerinta naudojant parametrų identifikavimą ir reguliatoriaus parametrų adaptavimą realiuoju laiku.

.

Bačys Justinas. INVESTIGATION OF IDENTIFICATION METHODS FOR TYPICAL PROCESS MODELS. Master's thesis in Control Systems / supervisor assoc. prof. dr. Rimvydas Simutis. Kaunas University of Technology, Faculty of Electrical and Electronics Engineering, department of Automation

Research area and field: Electrical and Electronics Engineering, Technological Sciences

Key words: identification, moving data window, closed loop control system.

Kaunas, 2017. 76 p.

SUMMARY

This thesis deals with analysis and investigation of the methods for processes identification that can be used to identify typical processes models from the data, obtained from normal functioning process.

The goal of this thesis is to investigate the typical processes model identification algorithms which can be used with the normal functioning data and to realize the adaptive control system using identified parameters of the model. In the first part of this investigation two stable identification methods was chosen and using all available processes data the influence of data sampling and noise level for identification quality was investigated.

In the second part of the thesis experiments with moving data window were conducted and quality of model parameters identification algorithms was estimated. In the end experiments with closed loop control system were conducted. In this control system the controller parameters were changed using on-line identified values of process model parameters. The quality of this system was compared with control system, where controller parameters are constant.

Perform experiments showed that the quality of the regulatory processes can be improved by using parameter identification algorithms and adaptation of the control parameters in real time.

TURINYS

SANTRUMPŲ IR TERMINŲ ŽODYNAS	8
ĮVADAS.....	9
1 LITERATŪROS ANALIZĖ	10
1.1 Procesų identifikavimas	10
1.2 Lėtų procesų su vėlavimų „on – line“ identifikavimas.....	11
1.3 Procesų identifikavimas realiu laiku, taikant neuro – fuzzy modelį.....	13
1.3.1 Fuzzy taisyklių apmokymas, taikant realaus laiko duomenis	14
1.3.2 Išplėstas Kalmano filtro apmokymo algoritmas.....	15
1.4 Identifikavimas, taikant diskretizuotas objekto įėjimo ir išėjimo reikšmes	16
1.5 Identifikavimas, taikant impulsines reakcijos kreives	17
1.5.1 Procesų modelio identifikavimas pagal impulsinę reakcijos kreivę.....	18
1.5.2 Perdavimo funkcijų parametru identifikavimo aspektai	19
1.5.3 Pirmosios eilės sistemos perdavimo funkcija.....	21
1.5.4 Antrosios eilės sistemos perdavimo funkcija	22
1.5.5 Antrosios eilės sistema su pirmosios eilės nuliu	22
1.5.6 Pirmosios eilės su vėlavimu sistema.....	23
2 MODELIŲ PARAMETRŲ IDENTIFIKAVIMAS PASITELKIANČIUS VISUS TURIMUS DUOMENIS	24
2.1 Pirmos eilės sistemos modelio identifikavimas.....	25
2.1.1 Sistemos modelio identifikavimas, kai signalas be triukšmo.....	27
2.1.2 Sistemos modelio identifikavimas, kai signalas su 2,5% triukšmu	30
2.1.3 Sistemos modelio identifikavimas, kai signalas su 5% triukšmu	33
2.1.4 Sistemos modelio identifikavimas, kai signalas su 10% triukšmu	36
2.2 Pirmos eilės sistemos modelio su vėlavimu identifikavimas	39
2.2.1 Sistemos modelio identifikavimas, kai signalas be triukšmo	41
2.2.2 Sistemos modelio identifikavimas, kai signalas su 2,5% triukšmu	45
2.2.3 Sistemos modelio identifikavimas, kai signalas su 5% triukšmu	49

2.2.4	Sistemos modelio identifikavimas, kai signalas su 10% triukšmu	53
3	PROCESO IDENTIFIKAVIMAS IR REGULIAVIMAS	57
3.1	Proceso identifikavimas iš slenkančio duomenų lango	58
3.1.1	Identifikavimas, taikant diskretizuotas objekto įėjimo ir išėjimo reikšmes	58
3.1.2	Identifikavimas, taikant netiesioginiu būdu suformuotas objekto impulsines reakcijos kreives	64
3.2	Proceso identifikavimas ir reguliavimas, naudojant slenkantį duomenų langą.....	65
3.2.1	Identifikavimas ir reguliavimas, taikant diskretizuotas objekto įėjimo ir išėjimo reikšmes	66
4	REZULTATAI IR IŠVADOS.....	75
5	LITERATŪROS ŠALTINIAI.....	76
	PRIEDAI	77

Santrumpų ir terminų žodynas

IAE (angl. Integral absolute error) –integruota absoliučioji paklaida.

Kalmano filtras – rekursinis filtras, kuris įvertina tiesinio dinaminio proceso būseną pagal modelio prognozę ir matavimus, kurie gaunami su triukšmu.

MAPE (angl. mean absolute percentage error) – vidutinė absoliutinė procentinė paklaida.

Neuro – fuzzy – hibridinė sistema, susidedanti iš neuroninio tinklo ir fuzzy modelio.

PI – reguliatorius susidedantis iš proporcinės (P) ir integralinės (I) dedamosios.

.

Įvadas

Procesų identifikavimas – tai proceso modelio sudarymas tik iš eksperimentinių įėjimo ir išėjimo duomenų. Naudojant šio tipo metodus netaikomi dėsniai, susiję su fundamentalia sistemos prigimtimi ir jos savybėmis.

Kadangi didelė sudėtingų procesų įvairovė ir ne visada galima apibrėžti juos, tad dinaminis procesas galima apibūdinti kaip „juoda dėžė“. (Tiksliai apibrėžti šiuos procesus yra sudėtinga dėl jų įvairovės, todėl šie dinaminiai procesai dažnai yra pavadinami „juodąja dėže.) Norint sudaryti tokiam procesui tinkamą matematinį modelį, taikomi proceso identifikavimo metodai, kuriais nustatomi svarbiausi proceso parametrai iš proceso reakcijos į tam tikrus įėjimo signalus. Tokie modeliai vadinami empiriniais.

Šio projekto tikslas atlikti tipinių procesų identifikavimo iš realaus funkcionavimo duomenų analizę ir realizuoti sistemos reguliavimą.

Uždaviniai:

1. atlikti literatūros apžvalgą apie procesų identifikavimo metodus, kuriuos galima taikyti siekiant identifikuoti tipinius procesus iš normalaus funkcionavimo duomenų;
2. pasirinkti stabilius identifikavimo metodus atlikti identifikavimo eksperimentus ir palyginti jų kokybę esant įvairiems parametrams;
3. atlikti identifikavimo eksperimentus su visais duomenimis;
4. atlikti identifikavimo eksperimentus su slenkančiu duomenų langu;
5. pritaikyti identifikavimą iš slenkančio duomenų lango regulatoriaus parametrų skaičiavimams;
6. atlikti eksperimentus reguliuojant sistemą pastoviais ir kintančiais regulatoriaus parametrais pagal identifikuotą procesą.

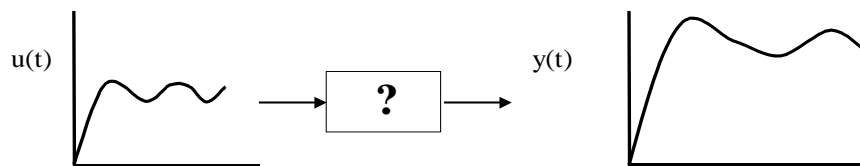
1 Literatūros analizė

Siekiant išsiaiškinti procesų identifikavimo metodus, reikia išsiaiškinti, kas yra identifikavimas ir kaip jis atliekamas. Tai bus išsamiai aptarta kitame poskyryje.

1.1 Procesų identifikavimas

Analizuojamas procesas yra apibūdinamas išmatuotais įėjimo $u(t)$ ir išėjimo $y(t)$ duomenimis. Jei nėra žinoma, pagal kokius dėsnius procesas reaguoja į įėjimo signalus, proceso modelį galima sudaryti, naudojant įėjimo ir išėjimo matavimų koreliacijas. Nežinomas procesas yra identifikuotas, jei nustatomos duomenų koreliacijos. Ši proceso modelio sudarymo metodika dažnai vadinama procesų identifikavimu.

Naudojant procesų identifikavimo (žr. 2.1 pav.) metodą nereikalingos jokios žinios apie proceso modelį. Nagrinėjamas procesas laikomas „juoda dėže“, kurios viduje procesams identifikuoti taikomi reakcijos į išorinį trikdį eksperimentiniai duomenys.



1.1 pav. Procesų identifikavimas

Identifikuojant procesą iš įėjimo ir išėjimo duomenų, praktikoje jis aprašomas tiesiniu perdavimo funkcijos modeliu, o norint sudaryti netiesinį modelį, išskiriami iš proceso keli darbo taškai, kuriuose identifikuojamas procesas aproksimuojamas kaip tiesinis.

Sudarant tinkamą proceso modelį, reikia sustabdyti normalų sistemos darbą ir atlikti su ja tinkamus eksperimentus. Tokia procedūra palengvina empirinio modelio sudarymą ir pagerina jo kokybę. Šis identifikavimas atliekamas taikant specialius istorinius proceso duomenis („angl. off – line“ - identifikavimas).

Procesų valdymui ir reguliavimui labiausiai paplitusios modelio formos [9]:

- pirmos eilės sistema su vėlavimu;
- antros eilės sistema su vėlavimu;
- antros eilės sistema su vėlavimu ir pirmos eilės nulių.

Atliekant modelių identifikavimą naudojamos įvairios funkcijos, kurių pagalba veikiamas objektas [10]:

- šuolinė funkcija;
- impulsinė funkcija;
- stačiakampis impulsas;
- sinusoidės;
- baltas triukšmas;
- atsitiktinė dvejetainių signalų seka.

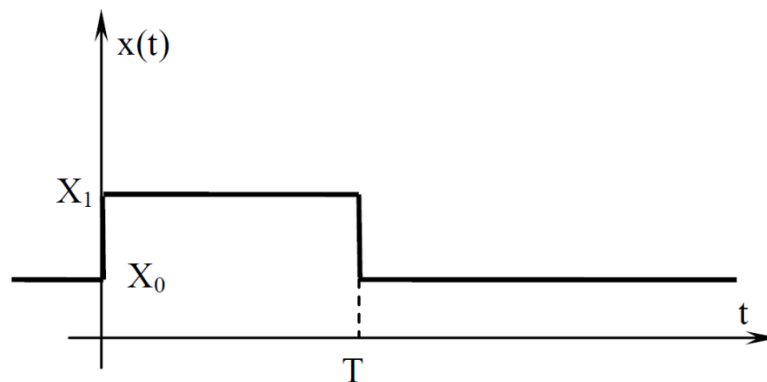
Naudojant šį metodą reikia pasirinkti tinkamą įėjimo ir išėjimo funkciją, kad iš proceso reakcijos būtų galima gauti kuo daugiau naudingos informacijos. Taip pat reikia nepamiršti, kad naudinga informacija gaunama iš išėjimo duomenų, tiesiogiai priklauso nuo įėjimo funkcijos.

Šio metodo taikymas gali daug kainuoti, nes nutraukiamas vykstantis procesas, o jo atstatymas gali užtrukti gana ilgą laiką. Tokiu atveju reikėtų naudoti procesų identifikavimą realiu laiku iš realių sistemos duomenų. Tokie metodai dar vadinami „angl. on – line“ - identifikavimu. Naudojant šį sistemos identifikavimo metodą, pakanka registruoti sistemos įėjimo ir išėjimo duomenis [1], [2].

1.2 Lėtų procesų su vėlavimu „on – line“ identifikavimas

„Angl on – line“ procesų identifikavimo metodo taikymą apibūdina sistemos, kurių balanso būseną apibūdina X_0 ir Y_0 įvesties signalas. Ryšys (1) apibūdina įvesties signalą, kuris pateiktas 2.2 paveiksle.[3].

$$x(t) = \begin{cases} X_1, & 0 \leq t \leq T_0 \\ X_0, & t > T_0 \end{cases} \quad (1)$$



1.2 pav. Įvesties signalas

Taikant šio tipo įvesties signalą, būsenos signalas turi du privalumus:

1. Galima pastebėti, kad Y išėjimas gali būti stabilizuotas tokia pačia stacionariame taške Y_0 . Jei Y_0 vertės negalima pasiekti, galima daryti išvadą, kad procesas, kuris gali atitikti modelį, nėra stacionarus arba atsirado dėl paklaidų darant eksperimentą, tad eksperimentas turi būti pakartotas.

2. Skirtumai $x(t) - X_0$ ir $y(t) - Y_0$ negalioja už laiko intervalo ribų, kai proceso stabilizavimo laiko intervalas yra didelis. Integralai:

$$I_{XK} = \int_0^{\infty} (-t)^K [x(t) - X_0] dt; \quad (2)$$

ir

$$I_{YK} = \int_0^{\infty} (-t)^K [y(t) - Y_0] dt \quad (3)$$

$K = 0, 1, \dots$, baigtinis skaičius.

Atsižvelgiant į aplinkybes, kad pasirinktas modelis priklauso nuo N parametru, proceso identifikavimas susideda iš $H(s)$ funkcijos nustatymo ir $N-1$ išvestinių. Rezultatas yra lygčių sistema su N lygčių ir N sistemos parametru.

$N-1$ išvestinės gali būti nustatomos atbuliniu būdu, naudojant priklausomybę

$$Y(s) = H(s) * X(s), \quad (4)$$

iš kurios gaunama

$$Y(0) = H(0) * X(0). \quad (5)$$

Sistemos perdavimo funkcijos išraiška:

$$H(s) = \frac{K e^{-\tau s}}{1 + Ts}, \text{ kur } T, \tau > 0. \quad (6)$$

Naudojant (4) priklausomybę galima gauti bendrą priklausomybę k – sios eilės išvestiniai lygtyje

$$H^{(K)}(0) = \frac{Y^{(K)}(0) - \sum_{i=1}^K C_i H^{(K-i)}(0) X^{(i)}(0)}{X(0)}, \quad (7)$$

kurioje

$$X^{(K)}(0) = \int_0^{\infty} (-t)^K [x(t) - X_0] dt \quad (8)$$

ir

$$Y^{(K)}(0) = \int_0^{\infty} (-t)^K [y(t) - Y_0] dt, \quad (9)$$

$K=0, 1, \dots, N-1$.

Integralas, kuris gaunamas iš (8) ir (9) išraiškų, gali būti skaičiuojamas taikant baigtinį intervalą $0 \leq t \leq T_0$, kur T_0 apibrezia integravimo ribas. Naudojant (6) modelio išraišką, gauta (7) išraiška ir priklausomybės:

$$\begin{aligned} H(0) &= K \\ H'(0) &= -K(T + \tau) \\ H''(0) &= K[(T + \tau)^2 + T^2] \end{aligned} \quad (10)$$

Naudojant šias priklausomybes, išvedami modelio parametrai:

$$\begin{aligned} H(0) &= K \\ T &= \sqrt{\frac{H''(0)}{H(0)} - \left(\frac{H'(0)}{H(0)}\right)^2} \\ \tau &= -\frac{H'(0)}{H(0)} - T \end{aligned} \quad (11)$$

1.3 Procesų identifikavimas realiu laiku, taikant neuro – fuzzy modelį

Sekant gaunamus naujus realius proceso duomenis, modelio identifikavimo procedūra automatiškai prasideda nuosekliai pritaikant neuro – fuzzy tinklo struktūrą ir parametrus naudojama priklausomybė (12) [6].

$$Y(k) = F_k(X_k) + G_k(X_k) * U(k), \quad (12)$$

$$F_k(X_k) = [f_{1k}(X_k), \dots, f_{bk}(X_k)]^T, \quad (13)$$

$$G_k(X_k) = \begin{bmatrix} g_{11k}(X_k) & \cdots & g_{1ak}(X_k) \\ \vdots & \ddots & \vdots \\ g_{b1k}(X_k) & \cdots & g_{bak}(X_k) \end{bmatrix}. \quad (14)$$

$Y(k)$ – apskaičiuotas išėjimo vektorius,

$U(k)$ – įėjimo vektorius.

$F_k(X_k)$, $G_k(X_k)$ – matricos sudarytos iš netiesinių funkcijų, aprašančių neuro – fuzzy tinklą.

Modelio identifikavimo procedūros etapai:

1. Rasti visus tinklo būseną apibūdinančius išėjimus.
2. Rasti artimiausią fuzzy taisyklę įėjimo duomenims $X(k)$.
3. Apskaičiuoti fuzzy taisyklės papildymo/pakeitimo kriterijų:

$$WRAR_{nr,j}(k) = \gamma_{1j} WRAR_{nr,j}(k-1) + \lambda_{nr,j}(k) \quad (15)$$

Šis kriterijus nurodo didžiausią svorį turinčią naujausią taisyklę, kuri paremta naujausiais praeities ir dabarties įvesties ir išvesties duomenimis, j – jam tinklo išėjimui. $\lambda_{nr,j}(k)$ rodo momentinį artimiausios fuzzy taisyklės įvertį j – jam tinklo išėjimui k – ju laiko momentu. Tai padeda įvertinti triukšmus veikiančius procesą [6].

$$\lambda_{nr,j}(k) = \frac{R_{nr,j}(X)}{\sum_{q=1}^{H_j} R_{qj}(X)} \quad (16)$$

Taip pat reikia įvertinti, ar apskaičiuota neuro – fuzzy struktūra yra adekvati. Tam naudojama $E_{WRMS,j}(k)$ įvertis.

$$E_{WRMS,j}(k) = \sqrt{\sum_{i=k-(N_j-1)}^k \gamma_{2j} \frac{i-k}{N_j} |e_j(i)|^2}, j = 1, \dots, b(a+1). \quad (17)$$

$e_j(i)$ – indikuojama j – ojo tinklo išėjimo paklaida, kuri apibūdina tinklo kokybę. N_j – duomenų langas j – ajame išėjime. γ_{2j} – žymi atitinkamą užmiršimo faktorių.

4. Patikrinti taisyklių papildymo/pakeitimo kriterijų.

Tikrinimo algoritmas:

$$\text{If} (||X(k) - \mu_{nr,j}|| > \varepsilon_j(k) \text{ and } E_{WRMS,j}(k) > E_{th,j})$$

If $WRAR_{nr,j}(k) > WRAR_{th,j}$

Priskiriama nauja taisyklė (H_j+1) su naujais parametrais:

$$a_{H_j+1,j} = e_j$$

$$\mu_{H_j+1,j} = X$$

$$\sigma_{H_j+1,j} = k_j ||X(k) - \mu_{nr,j}||$$

k_j – sutapimo faktorius nustatant taisyklės tinkamumą įėjime,

Else

Pakeisti tinkamiausioje taisyklėje parametrus:

$$a_{nr,j} = e_j$$

$$\mu_{nr,j} = X$$

$$\sigma_{nr,j} = k_j ||X(k) - \mu_{nr,j}||$$

End if

Else

Pakoreguoti artimiausios taisyklės parametrus ($a_{nr,j}, \mu_{nr,j}, \sigma_{nr,j}$), naudojant išplėstą Kalmano filtro mokymo algoritmą.

End if

5. Patikrinti taisyklių supaprastinimo kriterijus.

Jeigu $WRAR_{nr,j}(k) > WRAR_{p,j}$, tai reikia panaikinti artimiausią taisyklę ir sumažinti modelio parametrų vektorių.

$\varepsilon_j(k)$, $WRAR_{th,j}$ ir $E_{th,j}$ – apibūdina sprendimų ribas j – ojo modelio tinklo apskaičiavimui. $\varepsilon_j(k)$ – rodo minimalų naujausių ir didžiausią įtaką turinčių įvesties duomenų kiekį, kuris reikalingas netiesinėms fuzzy taisyklėms [6].

1.3.1 Fuzzy taisyklių apmokymas, taikant realaus laiko duomenis

Neuro – fuzzy modelio apmokymo procesas yra naujų fuzzy taisyklių ar jau egzistuojančių fuzzy taisyklių pakeitimas pagal naujus įvesties duomenis. Tačiau kai stebėjimo duomenys neperduoda naujumo kriterijaus, neuro – fuzzy tinklo parametrus, kurie yra būdingi taisyklių rinkiniui ir rezultatams, galima atnaujinti naudojant išplėstą Kalmano filtro apmokymo algoritmą. Esant kelių įėjimų modeliui, įvesties duomenų kiekis didėja ir tai gali sukelti sunkumų parametrų apmokymo procese. Tai atsitinka dėl to, kad didėjant įėjimų skaičiui, didėja ir taisyklių skaičius bei kovariacijos matrica. Dėl šių priežasčių išplėsto Kalmano filtro apmokymo algoritmas tampa didelis ir mokymas užima daug laiko, tad atsiranda skaičiavimų perkrova. Siekiant sumažinti šią problemą, apmokymo algoritme naujiems įvesties duomenims naudojama taisyklės laimėtojos strategija. Ši

strategija paremta tuo, kad išrenkama labiausiai tinkanti taisyklė įvesties duomenims ir ji panaudojama apmokymui [6].

1.3.2 Išplėstas Kalmano filtro apmokymo algoritmas

Išplėstas Kalmano filtro apmokymo algoritmas naudojamas netiesinių parametru įvertinimui. Pritaikius šį algoritmą neuro – fuzzy tinklui, šio algoritmo pagalba ieškomi tinklą apibūdinantys parametrai: $a_{nr.j}$, $\mu_{nr.j}$ ir $\sigma_{nr.j}$. Šiems parametrams rasti naudojamos išraiškos:

$$\dot{a}_{nr.j} = \frac{\partial \hat{y}_{af_j}(k)}{\partial a_{nr.j}} = \frac{R_{nr.j}}{\sum_{q=1}^{H_j} R_{qj}}, \quad (18)$$

$$\dot{\mu}_{nr.j} = \frac{\partial \hat{y}_{af_j}(k)}{\partial \mu_{nr.j}} = \frac{\partial \hat{y}_{af_j}(k)}{\partial R_{nr.j}} \frac{\partial R_{nr.j}}{\partial \mu_{nr.j}} = a_{nr.j} \left[\frac{\sum_{q=1}^{H_j} R_{qj} - R_{nr.j}}{\left(\sum_{q=1}^{H_j} R_{qj}\right)^2} \right] \frac{\partial R_{nr.j}}{\partial \mu_{nr.j}}, \quad (19)$$

$$\dot{\sigma}_{nr.j} = \frac{\partial \hat{y}_{af_j}(k)}{\partial \sigma_{nr.j}} = \frac{\partial \hat{y}_{af_j}(k)}{\partial R_{nr.j}} \frac{\partial R_{nr.j}}{\partial \sigma_{nr.j}} = a_{nr.j} \left[\frac{\sum_{q=1}^{H_j} R_{qj} - R_{nr.j}}{\left(\sum_{q=1}^{H_j} R_{qj}\right)^2} \right] \frac{\partial R_{nr.j}}{\partial \sigma_{nr.j}}, \quad (20)$$

$$\frac{\partial R_{nr.j}}{\partial \mu_{nr.j}} = 2R_{nr.j} \frac{X(k) - \mu_{nr.j}}{\sigma_{nr.j}^2}, \quad (21)$$

$$\frac{\partial R_{nr.j}}{\partial \sigma_{nr.j}} = 2R_{nr.j} \frac{\|X(k) - \mu_{nr.j}\|^2}{\sigma_{nr.j}^3}. \quad (22)$$

Gradientas nežinomų parametru vektoriuis apibūdinimui $\dot{\theta}_{nr.j} = [\dot{a}_{nr.j}, \dot{\mu}_{nr.j}, \dot{\sigma}_{nr.j}]^T$ gali būti panaudoti žemiau pateiktos išraiškos:

$$K_j(k) = P_{nr.j}(k-1) \dot{\theta}_{nr.j}(k) \left[R_j(k) + \dot{\theta}_{nr.j}^T(k) P_{nr.j}(k-1) + \dot{\theta}_{nr.j}(k) \right]^{-1}, \quad (23)$$

$$\theta_{nr.j}(k) = \theta_{nr.j}(k-1) + K_j(k) e_j(k), \quad (24)$$

$$P_{nr.j}(k) = \left[1 - K_j(k) \dot{\theta}_{nr.j}^T(k) \right] P_{nr.j}(k-1). \quad (25)$$

K_j – indikuoja Kalmano stiprinimą;

R_j – žymi modeliavimo paklaidos kovariaciją;

$P_{nr.j}$ – žymi parametru paklaidos kovariacijos matricą.

Visi parametrai susiję su artimiausia fuzzy taisykle, kuri leidžia rasti j – ojo išėjimo reikšmes [6].

1.4 Identifikavimas, taikant diskretizuotas objekto įėjimo ir išėjimo reikšmes

Šio proceso identifikavimo metodo esmė ta, kad duomenys gaunami diskretiniais laiko momentais $t_0, t_1, t_2, \dots, t_N$. Teorinio modelio reakcijos kreivės įverčiai tais pačiais laiko momentais $y(t_k), k=1, 2, \dots, N$ gaunami taikant išraišką, kuri priklauso nuo ieškomų parametrų:

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + b_1 u(k-m-1) + b_2 u(k-m-2). \quad (26)$$

Tokią išraišką galime gauti iš (27) išraiškos, atlikus tam tikras matematinės operacijas.

$$g(s) = \frac{K(\xi s + 1)e^{-\omega s}}{(\tau_1 s + 1)(\tau_2 s + 1)}, \quad (27)$$

Nustatant funkcijos $y(k)$ parametrus a_i ir b_i , jų išraiškos atitinkamai supaprastėja. Juos nustatyti galima naudojant mažiausių kvadratų metodą [5].

$$AB = (X^T X)^{-1} X^T Y \quad (28)$$

X matricos formavimas:

$$X = \begin{bmatrix} y(k-n) & y(k-n+1) & u(k-n) & u(k-n+1) \\ \vdots & \vdots & \vdots & \vdots \\ y(k-2) & y(k-3) & u(k-2) & u(k-3) \\ y(k-1) & y(k-2) & u(k-1) & u(k-2) \end{bmatrix} \quad (29)$$

Y matricos formavimas:

$$Y = \begin{bmatrix} y(k-n-1) \\ \vdots \\ y(k-1) \\ y(k) \end{bmatrix} \quad (30)$$

AB matricos rezultatas:

$$AB = \begin{bmatrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{bmatrix} \quad (31)$$

Radę a_i ir b_i , iš algebrinių lygčių vėliau randame originalios perdavimo funkcijos parametrus K, τ_1, τ_2 ir ξ .

Naudojant (32), (33), (34) ir (35) lygtis galima rasti perdavimo funkcijos parametrus.

$$a_1 = (e^{-\Delta t / \tau_1} + e^{-\Delta t / \tau_2}) \quad \text{ir} \quad a_2 = -(e^{-\Delta t / \tau_1} e^{-\Delta t / \tau_2}), \quad (32)$$

$$b_1 = K \left[1 - \left(\frac{\tau_1 - \xi}{\tau_1 - \tau_2} \right) e^{-\Delta t / \tau_1} - \left(\frac{\tau_2 - \xi}{\tau_2 - \tau_1} \right) e^{-\Delta t / \tau_2} \right], \quad (33)$$

$$b_2 = K \left[e^{-\Delta t / \tau_1} e^{-\Delta t / \tau_2} - \left(\frac{\tau_1 - \xi}{\tau_1 - \tau_2} \right) e^{-\Delta t / \tau_2} - \left(\frac{\tau_2 - \xi}{\tau_2 - \tau_1} \right) e^{-\Delta t / \tau_1} \right], \quad (34)$$

$$m = \frac{\alpha}{\Delta t}. \quad (35)$$

Vėlavimo laikas m lygtyje (26) yra randamas nepriklausomai nuo kitų kintamųjų, naudojant kovariaciją tarp U ir Y duomenų [4].

Pasinaudojus (32), (33) ir (34) lygtimis, nesunku gauti $y(k)$ išraiškas paprastesniems procesams, kurių perdavimo funkcijos aprašytos tokiomis lygtimis:

$$g(s) = \frac{K e^{-\alpha s}}{\tau s + 1}, \quad \tau = \tau_1 \tau_2 = 0.0, \quad \xi = 0.0, \quad (36)$$

$$g(s) = \frac{K e^{-\alpha s}}{(\tau_1 s + 1)(\tau_2 s + 1)}, \quad \xi = 0.0. \quad (37)$$

1.5 Identifikavimas, taikant impulsines reakcijos kreives

Procesų identifikavimo metodo pagal impulsines reakcijos kreives pagrindas yra proceso perdavimo funkcijos radimas, padavus impulsinį signalą į sistemos įėjimą. Tai galima daryti, nes egzistuoja tiesioginė priklausomybė tarp sistemos perdavimo funkcijos bei tos pačios sistemos reakcijos į impulsinį trikdį [1].

Naudojant šį metodą galima iš realių duomenų suformuoti impulsinę reakcijos kreivę. Formavimui reikalingi laisvai suformuoti įėjimo $u(t)$ ir išėjimo $y(t)$ duomenys. Jei sistema yra tiesinė arba artima tiesinei, tai tokios sistemos impulsinės funkcijos ir proceso reakcijos į įėjimo signalus priklausomybė aprašoma tokia išraiška:

$$y(k) = \sum_{i=1}^N g(i)u(k-i). \quad (38)$$

Stebėtų taškų išraiškos:

$$y(1) = g(1)u(0),$$

$$y(2) = g(1)u(1) + g(2)u(0), \quad (39)$$

$$y(3)=g(1)u(2)+g(2)u(1)+g(3)u(0),$$

t.t.

Sutvarkius išraiškas, gaunamos matricos:

$$\begin{bmatrix} y(1) \\ y(2) \\ \dots \\ y(M) \end{bmatrix} = \begin{bmatrix} u(0) & 0 & \dots & 0 \\ u(1) & u(0) & \dots & 0 \\ \dots & \dots & \dots & \dots \\ u(M-1) & u(M-2) & \dots & u(M-N) \end{bmatrix} \begin{bmatrix} g(1) \\ g(2) \\ \dots \\ g(N) \end{bmatrix}, \quad (40)$$

arba matricinė forma

$$y = U g. \quad (42)$$

N – bendras sistemos impulsinės reakcijos $g(t)$ taškų skaičius. Jie gaunami iš įėjimo ir išėjimo duomenų $u_k, y_k, k=1,2,\dots,M, N \leq M$. Priimama, kad reakcija į laisvai pasirinktą įėjimą yra tiesioginė vienietinės šuolinės funkcijos transformacija per „įėjimo matricą“ U .

Jei tenkinamos sąlygos: $M=N$ ir $u(0) \neq 0$, o matrica U yra kvadratinė ir neišsigimusi, tai (41) lygties sprendinys bus:

$$g = U^{-1}y. \quad (42)$$

Panaudojus šią transformaciją galima gauti impulsinės reakcijos kreivės duomenis $g(t), k=1, 2, \dots, N$ iš proceso reakcijos į laisvai formuojamą įėjimo signalą. Išplėsta sprendinio išraiška:

$$g(M) = \frac{1}{u(0)} [y(M) - g(1)u(M-1) - g(2)u(M-2) - \dots - g(M-1)u(1)]. \quad (43)$$

Šis metodas naudingas pramoninių procesų modelių kūrimo, kai negalima trikdyti proceso dėl technologinių apribojimų, kai proceso įėjimus formuoja ne modelį sudarantis valdymo inžinierius, o pramoninio proceso technologas [1].

1.5.1 Procesų modelio identifikavimas pagal impulsinę reakcijos kreivę

Kaip anksčiau buvo minėta, kad iš normalaus funkcionavimo duomenų galima suformuoti impulsinę reakcijos kreivę, tai iš impulsinės reakcijos kreivės galima rasti modelio parametrus.

Jei $g(s)$ yra sistemos perdavimo funkcija, tai tokios funkcijos atvirkštinė Laplaso transformacija bus identiška sistemos reakcijai į impulsinį signalą, $y(t) = g(t)$.

Pagal apibrėžimą funkcijos Laplaso transformacija yra:

$$g(s) = \int_0^{\infty} e^{-st} g(t) dt \quad (44)$$

Išskleidus e^{-st} eilute, pertvarkoma lygtis:

$$g(s) = \int_0^{\infty} \sum_{j=0}^{\infty} \frac{(-st)^j}{j!} g(t) dt \quad (45)$$

arba

$$g(s) = \int_0^{\infty} \sum_{j=0}^{\infty} (-1)^j \frac{s^j t^j}{j!} g(t) dt \quad (46)$$

Jei nagrinėjama procesui galima pritaikyti Laplaso transformaciją, išraiškos (45) suma turi konverguoti. Ši savybė leidžia pakeisti integravimo ir sumavimo tvarką lygtyje (46), tad ją galima pertvarkyti:

$$g(s) = \sum_{j=0}^{\infty} (-1)^j \frac{(s)^j}{j!} \int_0^{\infty} t^j g(t) dt \quad (47)$$

arba

$$g(s) = \sum_{j=0}^{\infty} (-1)^j \frac{(s)^j}{j!} m_j, \quad (48)$$

čia m_j yra j -tasis impulsinės reakcijos funkcijos $g(t)$ momentas:

$$m_j = \int_0^{\infty} t^j g(t) dt \quad (49)$$

Ryšį tarp sistemos perdavimo funkcijos ir sistemos reakcijos į impulsinį signalą funkcijos momentais apibūdina išraiška (48).

Iš eksperimentinių reakcijos duomenų, apskaičiavus momentus, galima identifikuoti perdavimo funkciją [1].

1.5.2 Perdavimo funkcijų parametrų identifikavimo aspektai

Perdavimo koeficientas gali būti nustatytas iš (50) išraiškos, jei sistema yra statinė ir jos perdavimo funkcija – $g(s)$.

$$K = \lim_{s \rightarrow 0} g(s) \quad (50)$$

Išraišką (50) išraiškoje vietoje $g(s)$ (44) išraišką, gauname:

$$K = \lim_{s \rightarrow 0} \int_0^{\infty} e^{-st} g(t) dt \quad (51)$$

Tokios analizuojamos sistemos, kurios įėjime yra vienetinis impulsas, nulinis momentas lygus perdavimo koeficientui:

$$K = \lim_{s \rightarrow 0} \int_0^{\infty} g(t) dt = m_0, \quad (52)$$

Turint tokias išraiškas, galima jas pritaikyti perdavimo funkcijos koeficientų normavimui ir gauti normuotą perdavimo funkciją, kurios perdavimo koeficientas – lygus vienetui.

$$g_n(s) = \frac{g(s)}{K} \quad (53)$$

Normuotų reakcijos į impulsą momentų išraiška:

$$\mu_j = \frac{m_j}{m_0} \quad (54)$$

Pritaikius tai, kad $K = m_0$ ir (48) išraiškoje m_j padalinus iš m_0 , gauname:

$$g_n(s) = \sum_{j=0}^{\infty} (-1)^j \frac{(s)^j}{j!} \mu_j \quad (55)$$

arba

$$g_n(s) = 1 - \mu_1 s + \frac{\mu_2}{2} s^2 - \frac{\mu_3}{6} s^3 + \dots \quad (56)$$

Kadangi (50) išraiškoje yra žinomi eksperimentiniai duomenys, tad juos galima susieti su nežinomais perdavimo funkcijų parametrais. Jei procesas aprašomas tokia n – tosios eilės perdavimo funkcija:

$$g(s) = \frac{K}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_2 s^2 + a_1 s^1 + 1}, \quad (57)$$

tai jos normuota perdavimo funkcija atrodys taip:

$$g_n(s) = \frac{1}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_2 s^2 + a_1 s^1 + 1}. \quad (58)$$

Sulyginus (56) ir (58) išraiškas gaunama (59) išraiška. Toliau atliekami daugybos veiksmai, sulyginami s laipsnių koeficientai ir gaunamos išraiškos, siejančios momentus ir perdavimo funkcijos parametrus (60) [1].

$$1 = (a_n s^n + a_{n-1} s^{n-1} + \dots + a_2 s^2 + a_1 s^1 + 1) \left(1 - \mu_1 s + \frac{\mu_2}{2} s^2 - \frac{\mu_3}{6} s^3 + \dots \right). \quad (59)$$

$$a_1 = \mu_1,$$

$$a_2 = a_1 \mu_1 - \frac{\mu_2}{2} = \mu_1^2 - \frac{\mu_2}{2}, \quad (60)$$

$$a_3 = a_2 \mu_1 - a_1 \frac{\mu_2}{2} + \frac{\mu_3}{6},$$

.....

1.5.3 Pirmosios eilės sistemos perdavimo funkcija

Pirmos eilės sistemos modelio perdavimo funkcijos išraiška:

$$g_n(s) = \frac{1}{\tau s + 1}. \quad (61)$$

Remiantis (59) išraiška atrodys taip:

$$1 = (\tau s + 1) \left(1 - \mu_1 s + \frac{\mu_2}{2} s^2 - \frac{\mu_3}{6} s^3 + \dots \right). \quad (62)$$

Toliau abiejose lygybės pusėse sulyginami įvairių s laipsnių koeficientai:

$$\mu_1 = \tau,$$

$$\mu_2 = 2\tau^2,$$

$$\mu_3 = 6\tau^3,$$

.....

$$\mu_n = n! \tau^n. \quad (63)$$

Iš (63) išraiškų galima gauti keletą laiko pastoviosios τ reikšmių, tačiau dėl paklaidų verčiau nenaudoti aukštesnių, nei trečios eilės momentų.

Taip pat kiekvienam momentui apskaičiuoti naudojami tie patys eksperimentiniai duomenys. Nors rasti τ gana tik pirmojo momento įverčio, tačiau iš kitų momentų taip pat galima gauti

informacijos. Vienas iš pavyzdžių yra tai, kad radus aukštesnės eilės momentus galima įvertinti, kiek analizuojamos sistemos dinamika yra artima idealiai pirmosios eilės sistemai [1].

1.5.4 Antrosios eilės sistemos perdavimo funkcija

Antros eilės normuota perdavimo funkcijos išraiška:

$$g_n(s) = \frac{1}{a_2 s^2 + a_1 s + 1}. \quad (64)$$

Remiantis (59) lygtimi, gauname:

$$1 = (a_2 s^2 + a_1 s + 1) \left(1 - \mu_1 s + \frac{\mu_2}{2} s^2 - \frac{\mu_3}{6} s^3 + \dots \right). \quad (65)$$

Toliau kaip ir pirmos eilės atveju, sulyginus koeficientus esant skirtingiems s laipsniams abiejose lygybės pusėse, gauname (66) išraiškas, kurių pakanka nežinomiems perdavimo funkcijos parametrams rasti. Kitus momentus galima panaudoti papildomiems parametru įverčiams rasti [1].

$$\begin{aligned} a_1 &= \mu_1 \\ a_2 &= \mu_1^2 - \frac{\mu_2}{2} \end{aligned} \quad (66)$$

1.5.5 Antrosios eilės sistema su pirmosios eilės nuliu

Normuotos antros eilės sistemos su pirmos eilės nuliu perdavimo funkcija:

$$g_n(s) = \frac{\xi s + 1}{a_2 s^2 + a_1 s + 1}. \quad (67)$$

Remiantis (59) lygtimi, gauname:

$$(\xi s + 1) = (a_2 s^2 + a_1 s + 1) \left(1 - \mu_1 s + \frac{\mu_2}{2} s^2 - \frac{\mu_3}{6} s^3 + \dots \right). \quad (68)$$

Tokios išraiškos yra trys nežinomi parametrai, kurie randami sudarius lygčių sistemą iš trijų lygčių. Lygtys gaunamos, sulyginus koeficientus esant skirtingiems s laipsniams abiejose lygties pusėse:

$$\begin{aligned}
a_1 - \mu_1 &= \xi \\
\frac{\mu_2}{2} - a_1\mu_1 + a_2 &= 0 \\
\frac{\mu_3}{6} - a_1\frac{\mu_2}{2} + a_2\mu_1 &= 0
\end{aligned} \tag{69}$$

Atlikus lygčių pertvarkymus, gaunamos nežinomų perdavimo funkcijos koeficientų išraiškos [1]:

$$a_1 = \frac{3\mu_1\mu_2 - \mu_3}{2\mu_1^2 - \mu_2}; \quad a_2 = a_1\mu_1 - \frac{\mu_2}{2}; \quad \xi = a_1 - \mu_1; \tag{70}$$

1.5.6 Pirmosios eilės su vėlavimu sistema

Naudojant impulsines reakcijos kreives, galima rasti ir pirmos eilės sistemos su vėlavimu parametrus. Tokios sistemos normuota perdavimo funkcija atrodo taip:

$$g_n(s) = \frac{e^{-\alpha s}}{\tau s + 1} \tag{71}$$

Remiantis (59) lygtimi, gauname:

$$e^{-\alpha s} = (\tau s + 1) \left(1 - \mu_1 s + \frac{\mu_2}{2} s^2 - \frac{\mu_3}{6} s^3 + \dots \right) \tag{72}$$

Įvedus eksponentinės funkcijos eilutę, gauname:

$$\left(1 - \alpha s + \frac{\alpha^2 s^2}{2!} - \frac{\alpha^3 s^3}{3!} + \dots \right) = (\tau s + 1) \left(1 - \mu_1 s + \frac{\mu_2}{2} s^2 - \frac{\mu_3}{6} s^3 + \dots \right) \tag{73}$$

Kaip ir nagrinėtoms perdavimo funkcijoms, sulyginami koeficientai esant skirtingiems s laipsniams ir gaunamos dvi lygtys, iš kurių randami du nežinomi parametrai:

$$\begin{aligned}
-\alpha &= \tau - \mu_1 \\
\frac{\alpha^2}{2} &= -\tau\mu_1 + \frac{\mu_2}{2}
\end{aligned} \tag{74}$$

Padarius pertvarkymus lygtyse ir įvedus naujus žymėjimus gaunama nauja išraiška [1]:

$$\mu_g = \mu_1 = \tau + \alpha \tag{75}$$

2 Modelių parametrų identifikavimas pasitelkiant visus turimus duomenis

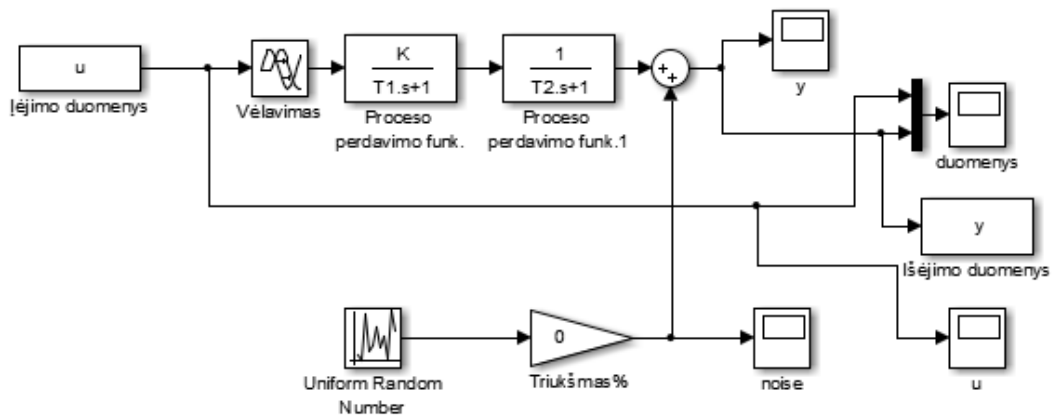
Tolimesnėse dalyse bus atliekami eksperimentiniai modelių identifikavimo tyrimai su normalaus funkcionavimo duomenimis pasitelkiant šiuos metodus:

1. Identifikavimas, taikant diskretizuotas įėjimo signalo ir objekto reakcijos kreivės reikšmes (a metodas).
2. Identifikavimas, taikant netiesioginiu būdu suformuotas objekto impulsines reakcijos kreives (b metodas).

Procesų identifikavimas visada pradedamas nuo paprastesnės sistemos (pirmos eilės sistema be vėlavimo) ir toliau įvertinama paklaida tarp realaus ir identifikuoto proceso. Jei rezultatai netenkina, pasirenkama sudėtingesnė sistema, kurios paklaida mažesnė.

Kadangi dažniausiai naudojamos pirmos eilės sistemos, tad tolimesniuose eksperimentiniuose tyrimuose bus naudojama pirmos eilės sistemos be vėlavimo ir su vėlavimu.

Procesų identifikavimui sukurtas modelis, naudojant „Matlab“ aplinką (žr. 3.1 pav.), kuris imituos procesą. Priklausomai nuo sistemos sudėtingumo, bus keičiami modelio parametrai. Modeliuojant procesą taip pat buvo aptriukšmintas proceso išėjimas tam, kad būtų patikrinta, kaip identifikavimo metodai gali nustatyti proceso duomenis iš triukšmingų duomenų.



2.1 pav. Proceso imitavimo modelis MATLAB/Simulink aplinkoje

Identifikuotam procesui ir jo parametrams įvertinti įvedama vidutinė absoliutinė procentinė paklaida (MAPE):

$$\delta = \left| \frac{k_{tik} - k_{id}}{k_{tik}} \right| * 100 \quad (76)$$

k_{tik} . – tikras parametras;

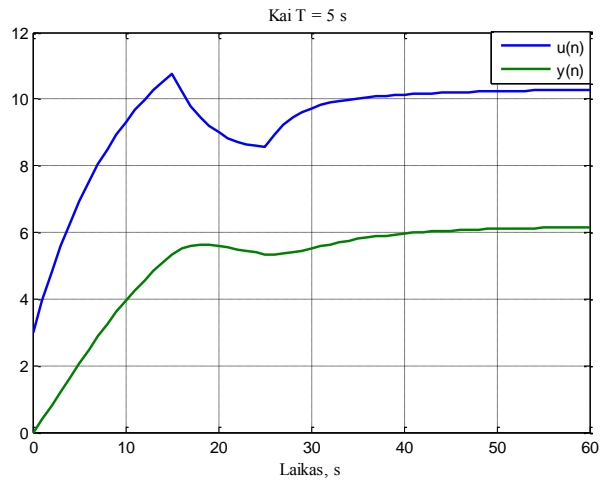
k_{id} . – identifikuotas parametras.

2.1 Pirmos eilės sistemos modelio identifikavimas

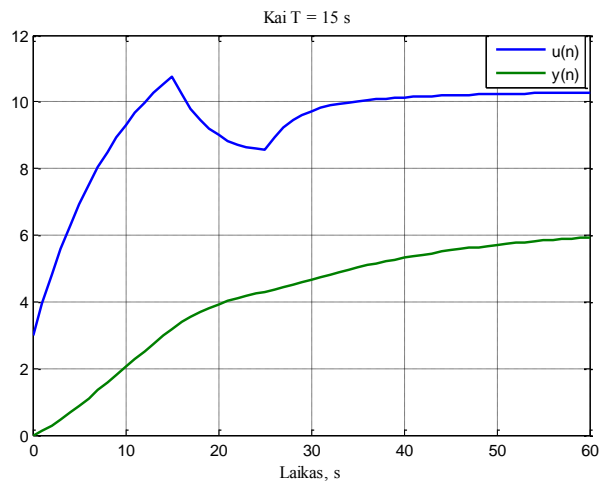
Naudojant a ir b identifikavimo metodus, atlikti pirmos eilės objekto be vėlavimo parametrų identifikavimo eksperimentai, kurių metu keičiamas duomenų diskretiškumas ir triukšmo lygis. Tolimesniuose poskyriuose pateikti eksperimento rezultatai.

Pirmos eilės objekto perdavimo funkcija naudojama tyrimui:

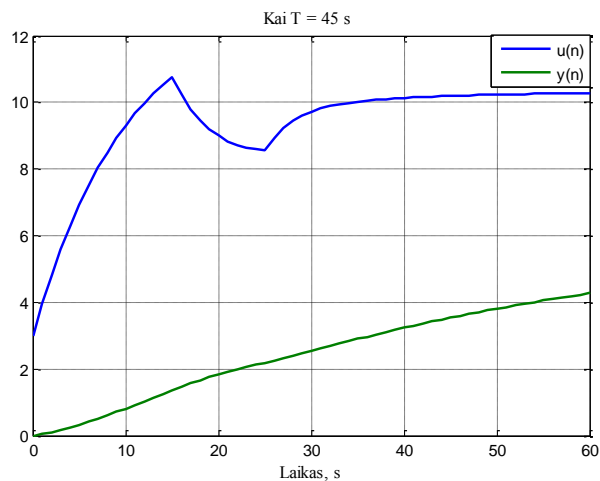
$$W(s) = \frac{K}{Ts+1} \quad (77)$$



2.2 pav. Sistemos įėjimo ir išėjimo duomenys grafiko pavidalu, kai $T = 5$ s



2.3 pav. Sistemos įėjimo ir išėjimo duomenys grafiko pavidalu, kai $T = 15$ s



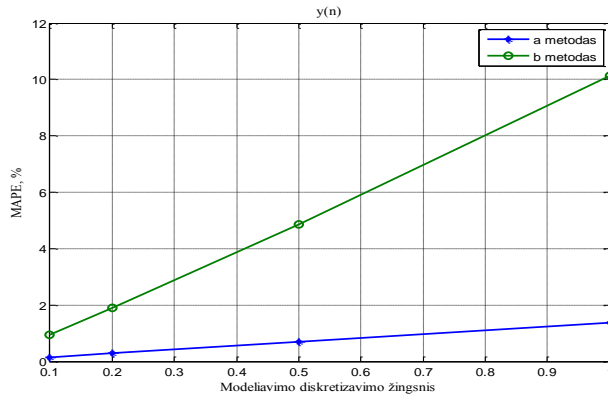
2.4 pav. Sistemos įėjimo ir išėjimo duomenys grafiko pavidalu, kai $T = 45s$
 Proceso parametrams rasti naudojamoms pateiktos „Matlab“ programos:

1. „Matlab“ aplinkos kodas, skirtas pirmos eilės sistemos be vėlavimo parametrams identifikuoti, taikant diskretizuotas įėjimo ir išėjimo reikšmes, pateiktas 1 priede.
2. „Matlab“ aplinkos kodas, skirtas pirmos eilės sistemos be vėlavimo parametrams identifikuoti, taikant dirbtinai suformuotas impulsinės reakcijos kreives, pateiktas 2 priede.

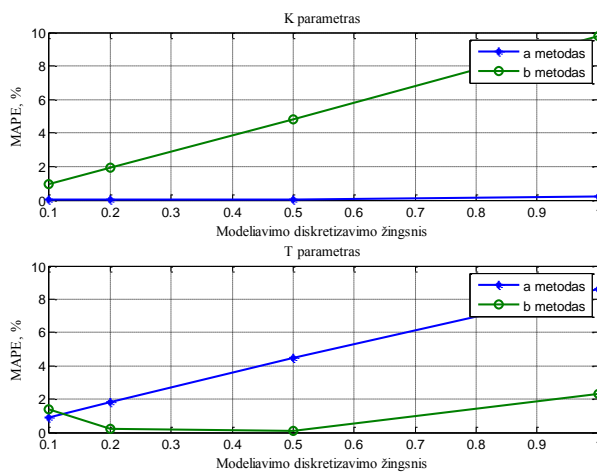
2.1.1 Sistemos modelio identifikavimas, kai signalas be triukšmo

2.1 lentelė. Tyrimo metu gauti rezultatai

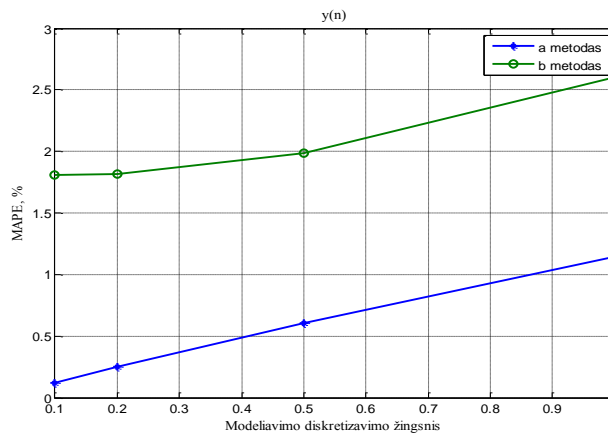
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$				$\Delta t=0,2s$				$\Delta t=0,5s$				$\Delta t=1s$			
Identifikuojami parametrai	K		T		K		T		K		T		K		T	
Parametras	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$
Sistemos parametrai	K=0,6; T=5 s															
a metodas	0,599 9	0,016 7	4,954 8	0,904	0,599 7	0,05	4,910 2	1,796	0,599 2	0,033 3	4,774 9	4,502	0,598 8	0,2	4,570 1	8,502
b metodas	0,594 3	0,95	4,993	1,4	0,588 5	1,917	4,989 8	0,204	0,571	4,833 3	5,003 5	0,07	0,541 4	9,767	5,116 3	2,326
Sistemos parametrai	K=0,6; T=15 s															
a metodas	0,599 6	0,066 7	14,93 59	0,427 3	0,599 3	0,116 7	14,87 31	0,846	0,598 3	0,283 3	14,69 36	2,042 7	0,596 8	0,533 3	14,42 25	3,84
b metodas	0,587 1	2,15	13,85 18	7,654 7	0,585 1	2,483 3	13,82 47	7,835 3	0,579	3,5	13,75 39	8,307 3	0,568 1	5,136 7	13,67 07	8,862
Sistemos parametrai	K=0,6; T=45 s															
a metodas	0,598 9	0,183 3	44,82 02	0,399 6	0,597 9	0,35	44,64 5	0,788 9	0,594 9	0,85	44,14 62	1,897 3	0,590 5	1,583 3	43,40 25	3,55
b metodas	0,441 2	26,46 67	23,47 37	47,83 62	0,440 5	26,58 33	23,42 57	47,94 29	0,438 3	26,95	23,28 84	48,24 8	0,434 3	27,61 67	23,08 1	48,70 89



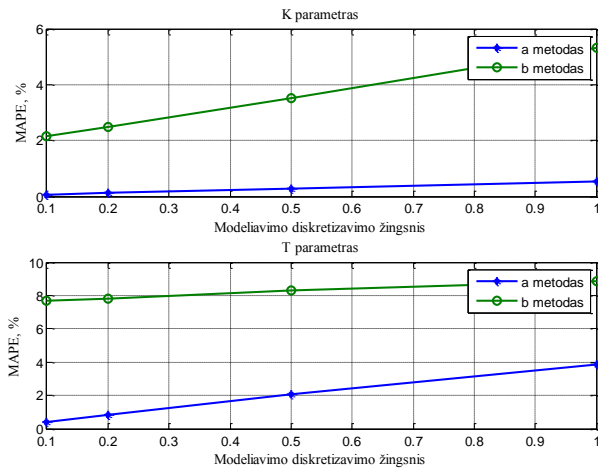
2.5 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



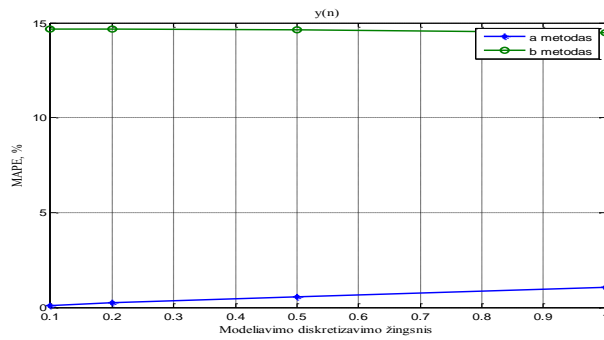
2.6 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėm absoliutinėm paklaidoms, kai $T = 5s$



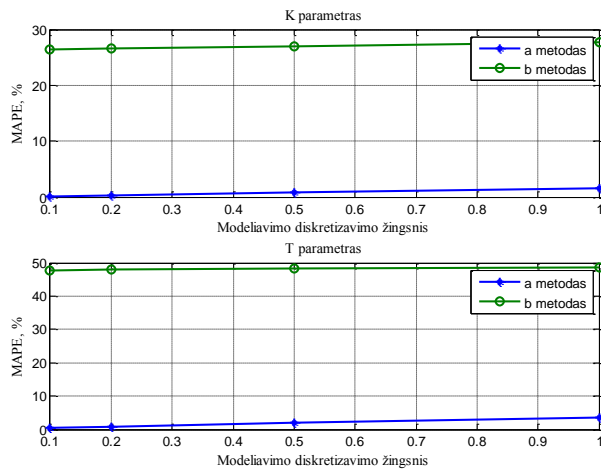
2.7 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.8 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidom, kai $T = 15s$



2.9 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



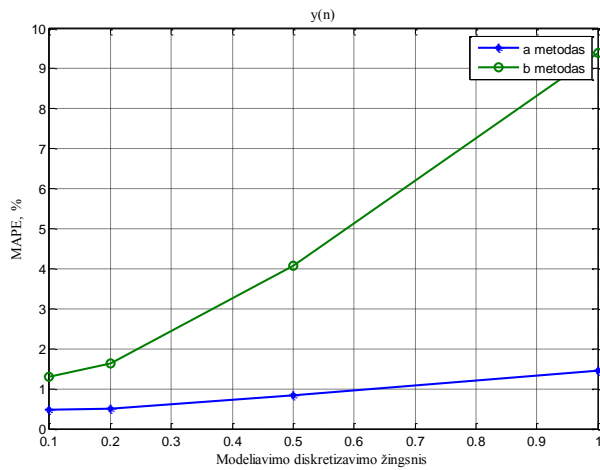
2.10 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidom, kai $T = 45s$

Atliekant eksperimentą pastebėta, kad didinant diskretizavimo žingsnį mažėja identifikuotų parametų tikslumas ir a metodas visais atvejais buvo tikslesnis.

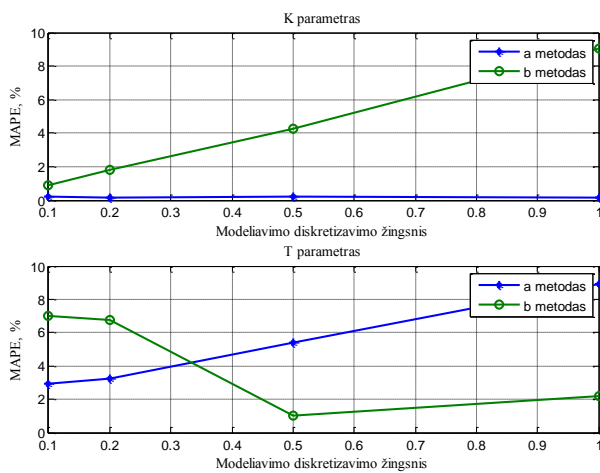
2.1.2 Sistemos modelio identifikavimas, kai signalas su 2,5% triukšmu

2.2 lentelė. Tyrimo metu gauti rezultatai

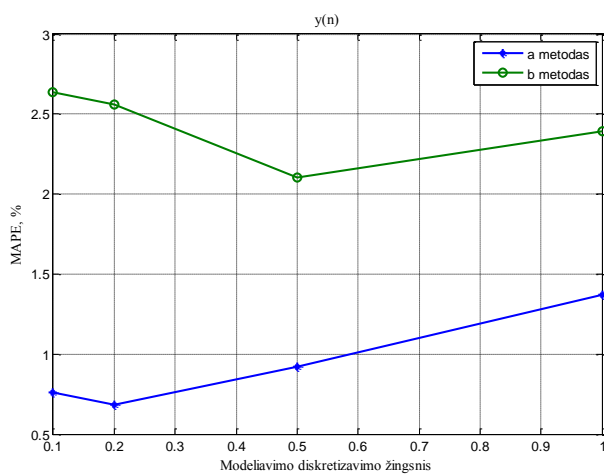
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$				$\Delta t=0,2s$				$\Delta t=0,5s$				$\Delta t=1s$			
	K		T		K		T		K		T		K		T	
Parametras	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$
Sistemos parametrai	K=0,6; T=5 s															
a metodas	0,598 7	0,216 7	4,853 4	2,932	0,599 0	0,166 7	4,838 6	3,228	0,598 8	0,2	4,731 1	5,378	0,599 1	0,15	4,552 7	8,946
b metodas	0,594 6	0,9	4,650 3	6,994	0,589 2	1,8	4,662 8	6,744	0,574 5	4,25	4,950 2	0,996	0,545 7	9,05	5,110 9	2,218
Sistemos parametrai	K=0,6; T=15 s															
a metodas	0,595 0	0,833 3	14,51 53	3,231 3	0,596 0	0,666 7	14,57 55	2,83	0,595 6	0,733 3	14,45 93	3,604 7	0,597 5	0,416 7	14,38 26	4,116
b metodas	0,587 4	2,1	13,49 98	10,00 13	0,585 8	2,366 7	13,48 52	10,09 87	0,582 5	2,916 7	13,64 88	9,008	0,572 4	4,6	13,60 07	9,328 7
Sistemos parametrai	K=0,6; T=45 s															
a metodas	0,562 4	6,266 7	40,08 05	10,93 22	0,571 7	4,716 7	41,28 93	8,246	0,574 6	4,233 3	41,59 28	7,571 6	0,593 0	1,166 7	43,32 07	3,731 8
b metodas	0,441 5	26,41 67	22,99 79	48,89 36	0,441 2	26,46 67	22,95 95	48,97 89	0,441 8	26,36 67	23,07 44	48,72 36	0,438 6	26,9	22,89 66	49,11 87



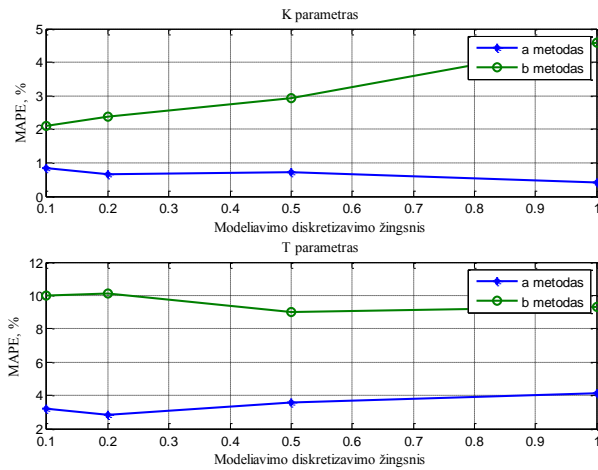
2.11 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



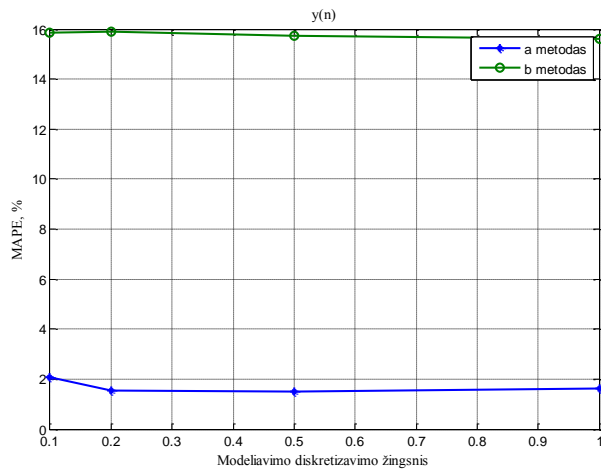
2.12 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 5s$



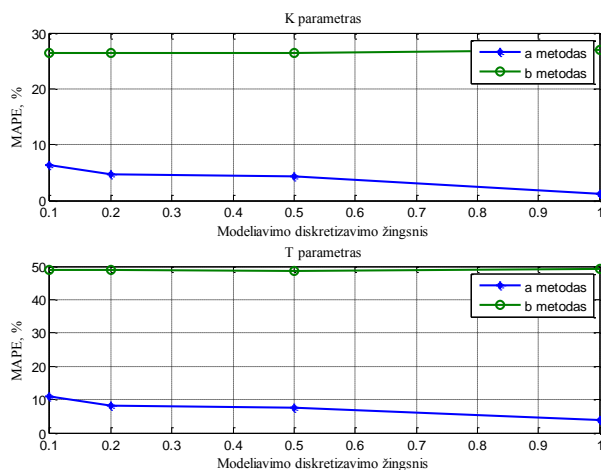
2.13 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.14 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 15s$



2.15 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



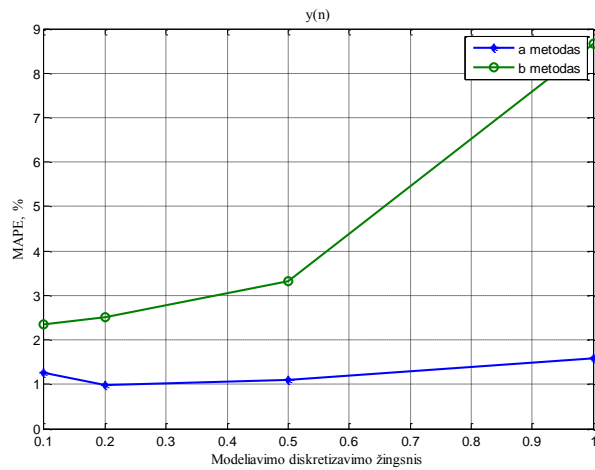
2.16 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 45s$

Atliekant eksperimentą prie modelio išėjimo duomenų buvo pridėtas triukšmas ir pastebėta, kad tarp diskretizavimo žingsnio ir vidutinių absoliutinių paklaidų nebėra tiesinės priklausomybės. Šiuo atveju a metodas buvo tikslesnis nei b metodas.

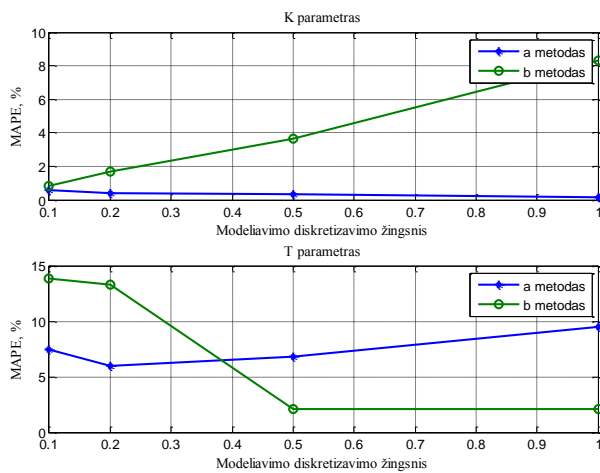
2.1.3 Sistemos modelio identifikavimas, kai signalas su 5% triukšmu

2.3 lentelė. Tyrimo metu gauti rezultatai

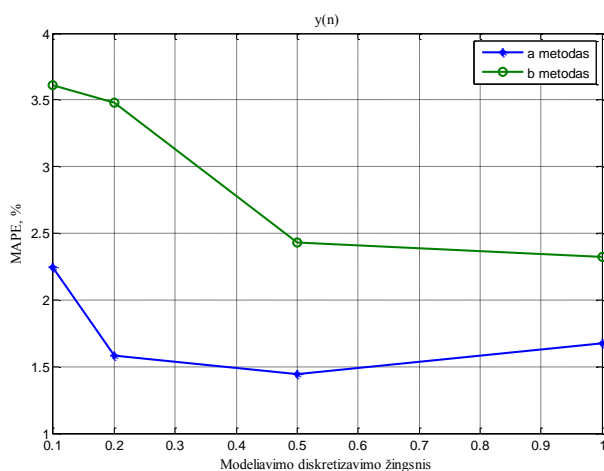
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$				$\Delta t=0,2s$				$\Delta t=0,5s$				$\Delta t=1s$			
	K		T		K		T		K		T		K		T	
Identifikuojami parametrai	K		T		K		T		K		T		K		T	
Parametras	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s$															
a metodas	0,5965	0,5833	4,6253	7,494	0,5977	0,3833	4,6998	6,004	0,5981	0,3167	4,6578	6,844	0,5992	0,1333	4,5233	9,534
b metodas	0,5950	0,8333	4,3080	13,84	0,5899	1,6833	4,3366	13,268	0,5780	3,6667	4,8975	2,05	0,5500	8,3333	5,1056	2,112
Sistemos parametrai	$K=0,6; T=15 s$															
a metodas	0,5859	2,35	13,625 3	9,1647	0,5903	1,6167	14,023 4	6,5107	0,5920	1,3333	14,127 1	5,8193	0,5977	0,3833	14,289 5	4,7367
b metodas	0,5878	2,0333	13,148 1	12,346	0,5865	2,25	13,146 4	12,357 3	0,5860	2,3333	13,544 9	9,7007	0,5768	3,8667	13,531 7	9,7887
Sistemos parametrai	$K=0,6; T=45 s$															
a metodas	0,5013	16,45	31,971 2	28,952 9	0,5298	11,7	35,772 8	20,504 9	0,5481	8,65	38,170 8	15,176	0,5900	1,6667	42,531 1	5,4864
b metodas	0,4418	26,366 7	22,522 9	49,949 1	0,4419	26,35	22,494 7	50,011 8	0,4452	25,8	22,863 7	49,191 8	0,4429	26,183 3	22,715 8	49,520 4



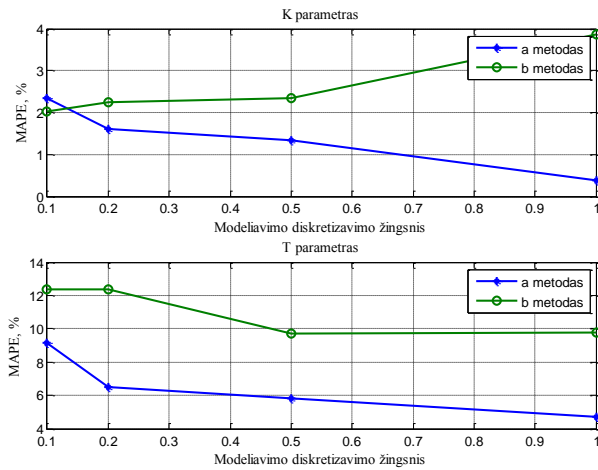
2.17 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



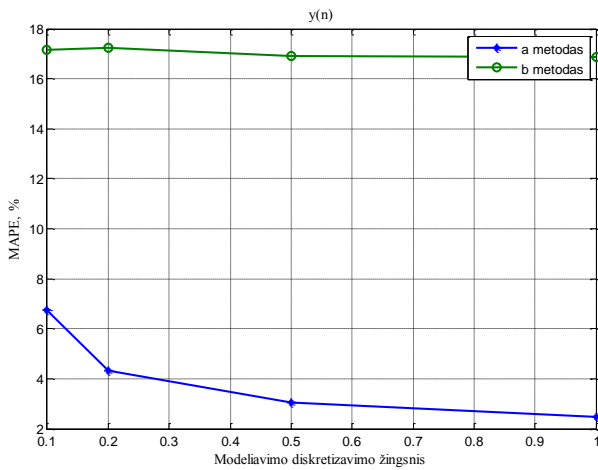
2.18 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 5s$



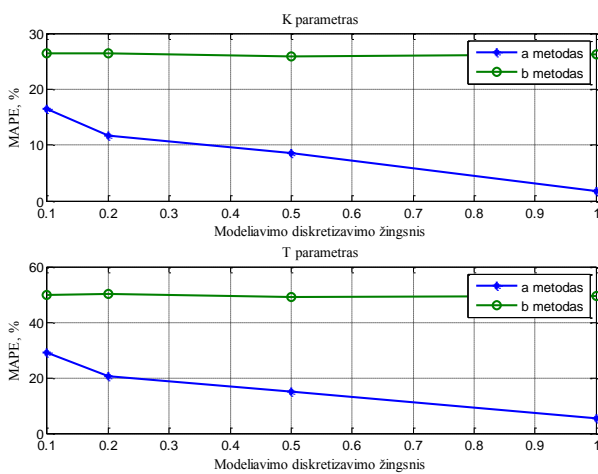
2.19 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.20 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 15s$



2.21 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



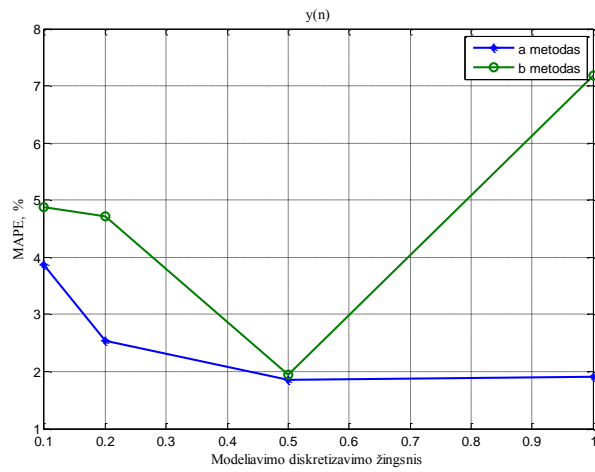
2.22 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 45s$

Atliekant eksperimentą pastebėta, kad didinant triukšmą reikėtų rinktis didesnę diskretizavimo žingsnį, kuris leidžia išvengti triukšmo įtakos. Didinant triukšmą matyti, kad naudojant a metodą parametrų identifikavimo kokybė vis dar geresnė nei naudojant b metodą.

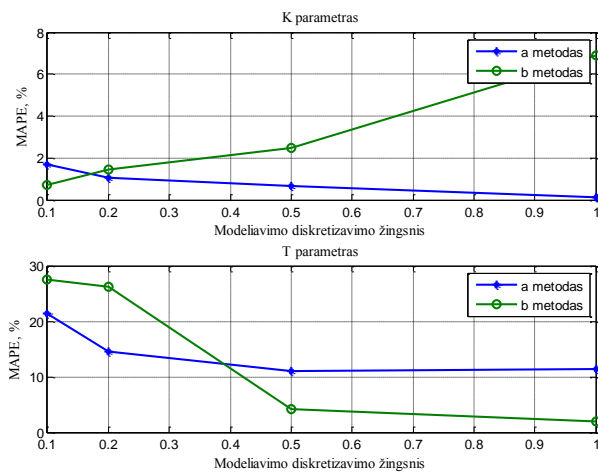
2.1.4 Sistemos modelio identifikavimas, kai signalas su 10% triukšmu

2.4 lentelė. Tyrimo metu gauti rezultatai

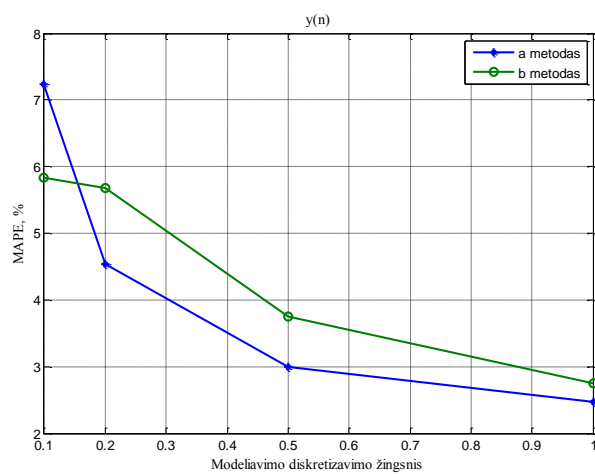
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$				$\Delta t=0,2s$				$\Delta t=0,5s$				$\Delta t=1s$			
Identifikuojami parametrai	K		T		K		T		K		T		K		T	
Parametras	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s$															
a metodas	0,589 9	1,683 3	3,932 0	21,36	0,593 7	1,05	4,269 2	14,61 6	0,596 1	0,65	4,445 9	11,08 2	0,599 2	0,133 3	4,430 7	11,38 6
b metodas	0,595 6	0,733 3	3,624 5	27,51	0,591 3	1,45	3,686 4	26,27 2	0,585 0	2,5	4,794 0	4,12	0,558 7	6,883 3	5,095 2	1,904
Sistemos parametrai	$K=0,6; T=15 s$															
a metodas	0,560 6	6,566 7	11,09 98	26,00 13	0,573 8	4,366 7	12,39 24	17,38 4	0,582 6	2,9	13,22 62	11,82 53	0,596 5	0,583 3	13,95 51	6,966
b metodas	0,588 5	1,916 7	12,44 61	17,02 6	0,588 0	2	12,47 14	16,85 73	0,593 0	1,166 7	13,34 09	11,06 07	0,585 4	2,433 3	13,39 69	10,68 73
Sistemos parametrai	$K=0,6; T=45 s$															
a metodas	0,398 2	33,63 33	18,16 76	59,62 76	0,443 7	26,05	24,29 51	46,01 09	0,489 4	18,43 33	30,47 80	32,27 11	0,570 1	4,983 3	39,19 86	12,89 2
b metodas	0,442 5	26,25	21,57 49	52,05 58	0,443 3	26,11 67	21,56 97	52,06 73	0,452 2	24,63 33	22,45 22	50,10 62	0,451 6	24,73 3	22,36 47	50,30 07



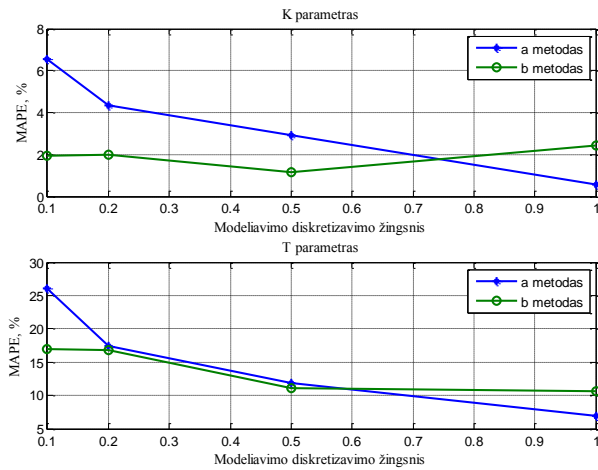
2.23 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



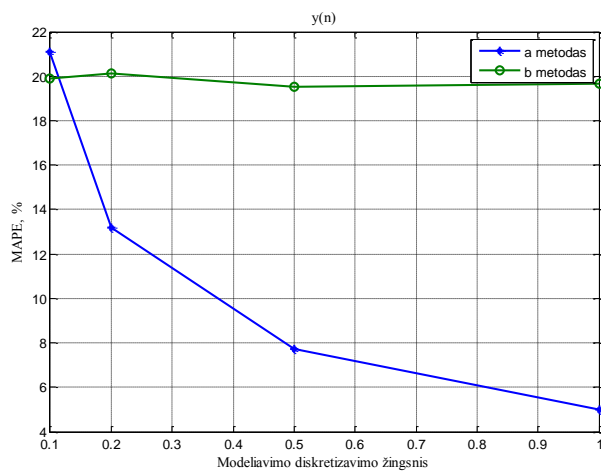
2.24 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 5s$



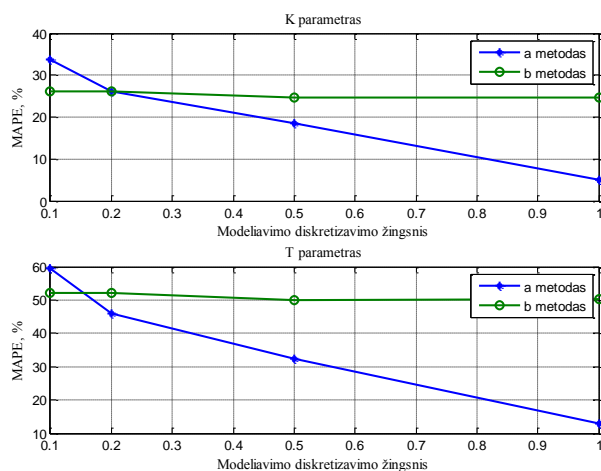
2.25 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.26 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 15s$



2.27 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



2.28 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 45s$

Atlikus eksperimentą pastebėta, kad esant dideliam signalo triukšmui reikėtų rinktis didesnę diskretizavimo žingsnį, kuris pagerina identifikavimo kokybę.

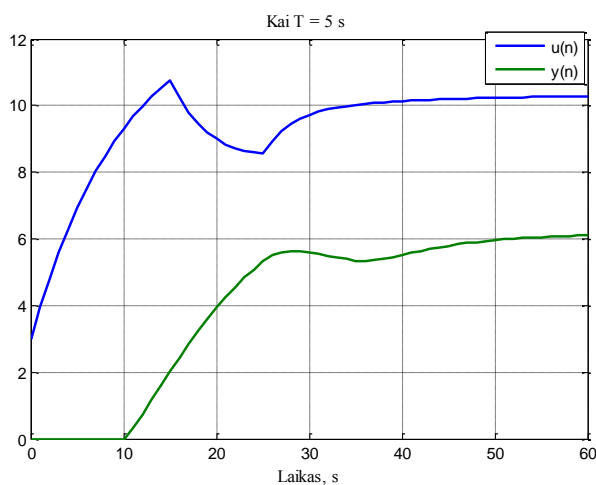
2.2 Pirmos eilės sistemos modelio su vėlavimu identifikavimas

Naudojant a ir b identifikavimo metodus, atlikti pirmos eilės objekto su vėlavimo parametru identifikavimo eksperimentai, kurių metu keičiamas duomenų diskretiškumas ir triukšmo lygis. Tolimesniuose poskyriuose pateikti eksperimento rezultatai.

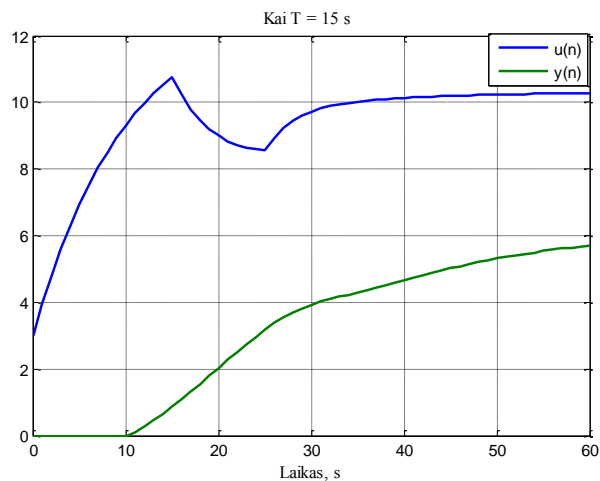
Pirmos eilės objekto perdavimo funkcija naudojama tyrimui:

$$W(s) = \frac{K e^{-\tau s}}{Ts+1} \quad (78)$$

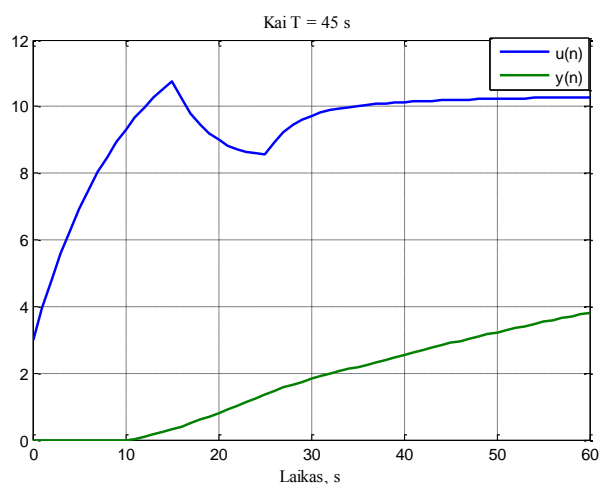
Kadangi taikant impulsinės reakcijos kreives pirmos eilės procesui su vėlavimu yra vėlavimo nustatymo metodika, tai ją galima realizuoti „Matlab“ aplinkos programos kode. Tačiau kitas metodas neturi vėlavimo identifikavimo, tad pasinaudota „Matlab“ funkcija vėlavimo nustatymui (finddelay(u,y)) [7].



2.29 pav. Sistemos įėjimo ir išėjimo duomenys grafiko pavidalu, kai $T = 5$ s



2.30 pav. Sistemos įėjimo ir išėjimo duomenys grafiko pavidalu, kai $T = 15$ s



2.31 pav. Sistemos įėjimo ir išėjimo duomenys grafiko pavidalu, kai $T = 45$ s

Proceso parametrams rasti naudojamos „Matlab“ programos:

1. „Matlab“ aplinkos kodas, skirtas pirmos eilės sistemos su vėlavimu parametrams identifikuoti, taikant diskretizuotas įėjimo ir išėjimo reikšmes, pateiktas 3 priede.
2. „Matlab“ aplinkos kodas, skirtas pirmos eilės sistemos su vėlavimu parametrams identifikuoti, taikant dirbtinai suformuotas impulsinės reakcijos kreives, pateiktas 4 priede.

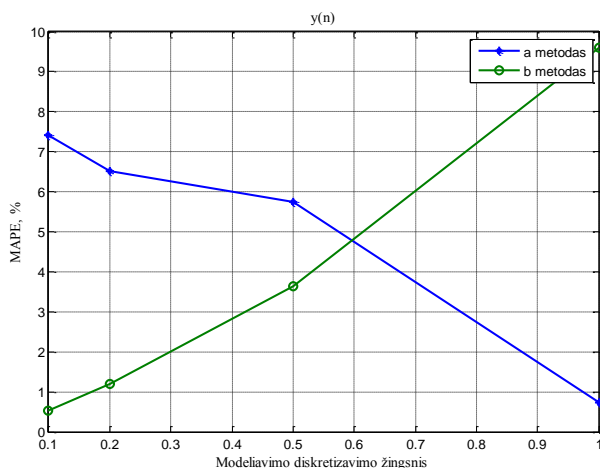
2.2.1 Sistemos modelio identifikavimas, kai signalas be triukšmo

2.5 lentelė. Tyrimo metu gauti rezultatai

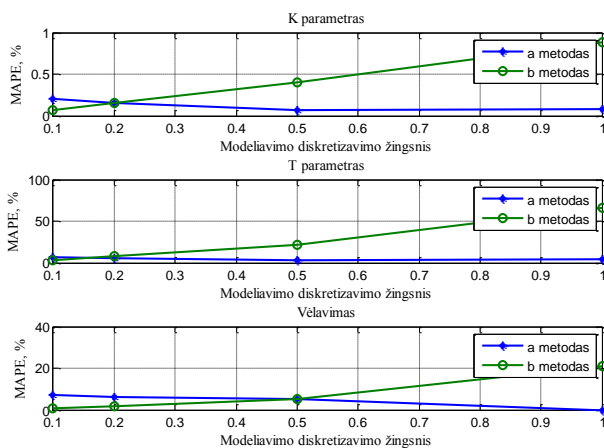
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$						$\Delta t=0,2s$					
Identifikuojami parametrai	K		T		τ		K		T		τ	
Parametras	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau=10 s;$											
a metodas	0,6012	0,2	5,3666	7,332	9,3	7	0,6009	0,15	5,2774	5,548	9,4	6
b metodas	0,5996	0,0667	4,8156	3,688	10,0714	0,714	0,5991	0,15	4,6239	7,522	10,15	1,5
Sistemos parametrai	$K=0,6; T1=15 s; \tau=10 s;$											
a metodas	0,6046	0,7667	15,8599	5,7327	8	20	0,6045	0,75	15,8414	5,6093	8	20
b metodas	0,5785	3,5833	11,3403	24,398	11,7232	17,232	0,5783	3,6167	11,3179	24,5473	11,6584	16,584
Sistemos parametrai	$K=0,6; T1=45 s; \tau=10 s;$											
a metodas	0,6211	3,5167	48,5380	7,8622	7,2	28	0,6209	3,4833	48,5001	7,778	7,2	28
b metodas	0,4024	32,9333	13,9910	68,9089	16,3860	63,86	0,4023	32,95	13,9791	68,9353	16,3118	63,118,

2.6 lentelė. Tyrimo metu gauti rezultatai

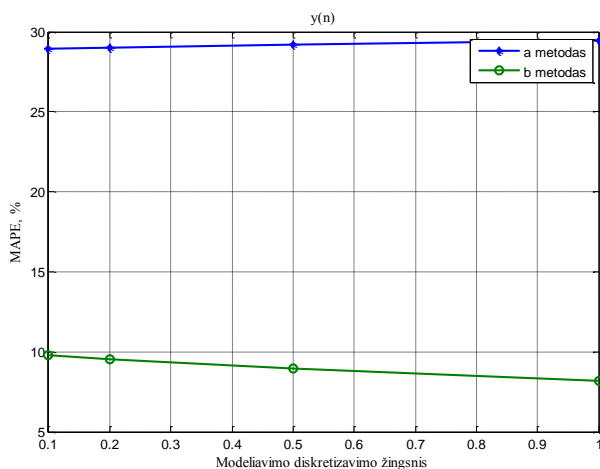
Duomenų nuskaitymo intervalas	$\Delta t=0,5s$						$\Delta t=1$					
Identifikuojami parametrai	K		T		τ		K		T		τ	
Parametras	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau =10 s;$											
a metodas	0,6004	0,0667	5,1417	2,834	9,5	5	0,5995	0,0833	4,8049	3,902	10	0
b metodas	0,5976	0,4	3,9044	21,912	10,5157	5,157	0,5947	0,8833	1,6798	66,404	12,1087	21,087
Sistemos parametrai	$K=0,6; T1=15 s; \tau =10 s;$											
a metodas	0,6042	0,7	15,7898	5,2653	8	20	0,6039	0,65	15,7166	4,7773	8	20
b metodas	0,5779	3,6833	11,2464	25,024	11,4657	14,657	0,5772	3,8	11,1173	25,8847	11,15	11,5
Sistemos parametrai	$K=0,6; T1=45 s; \tau =10 s;$											
a metodas	0,6164	2,733	47,7623	6,1384	7,5	25	0,6101	1,6833	46,7242	3,8316	8	20
b metodas	0,4019	33,0167	13,9431	69,0153	16,0890	60,89	0,4013	33,1167	13,8824	69,1502	15,7173	57,1730



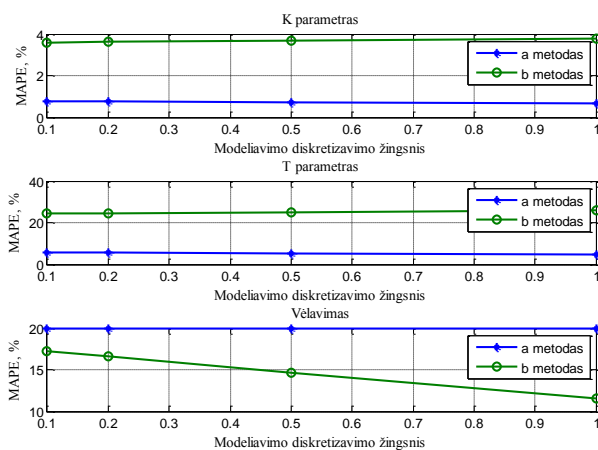
2.32 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



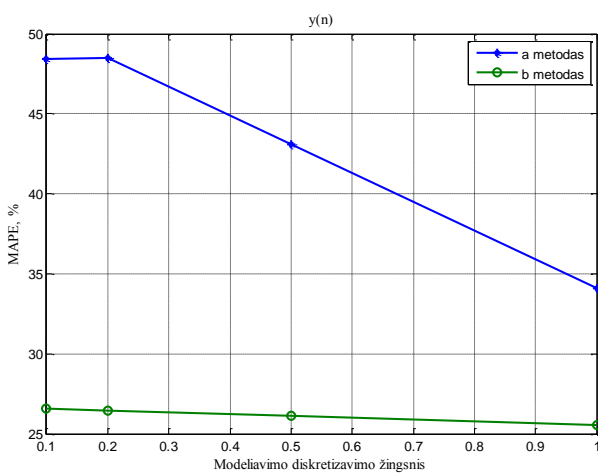
2.33 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 5s$



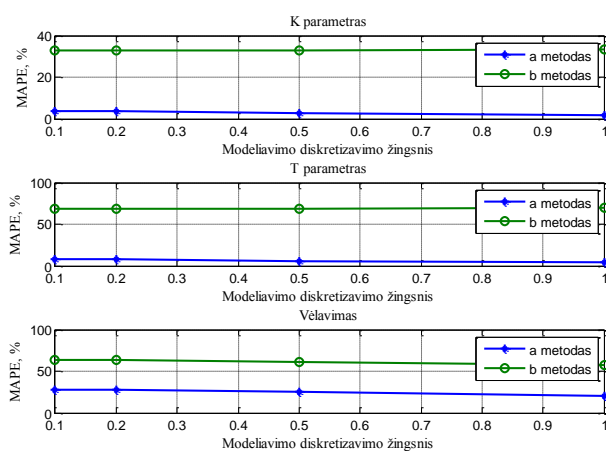
2.34 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.35 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 15s$



2.36 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



2.37 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 45s$

Atliekant eksperimentus su vėlavimu pastebėta, kad parametų identifikavimo kokybė blogėja, kadangi a metodas neturi vėlavimo identifikavimo metodikos ir pasirinkus blogą vėlavimo identifikavimo metodą blogėja visų parametų identifikavimas.

Naudojant b metodą parametų identifikavimo kokybė blogesnė nei a metodo, tačiau modelio su identifikuotais parametrais vidutinė absoliutinė paklaida mažesnė.

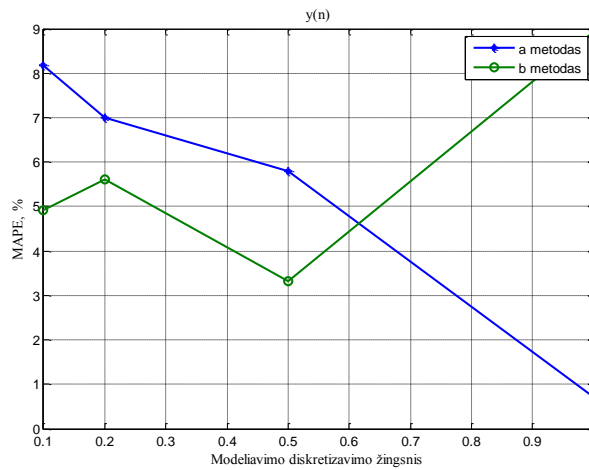
2.2.2 Sistemos modelio identifikavimas, kai signalas su 2,5% triukšmu

2.7 lentelė. Tyrimo metu gauti rezultatai

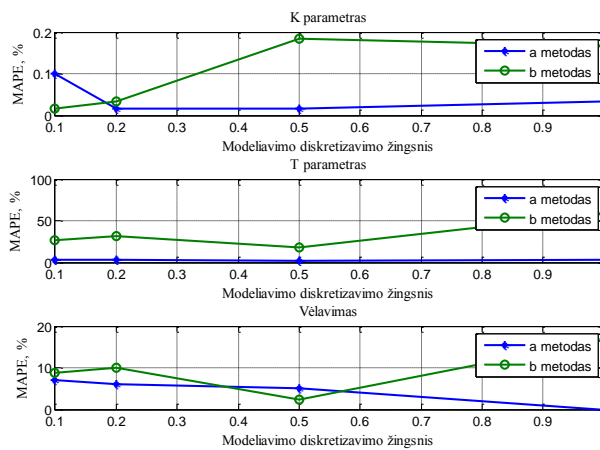
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$						$\Delta t=0,2s$					
	K		T		τ		K		T		τ	
Parametras	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau=10 s;$											
a metodas	0,5994	0,1	5,1359	2,718	9,3	7	0,5999	0,0167	5,1329	2,658	9,4	6
b metodas	0,5999	0,0167	3,6743	26,514	10,8674	8,674	0,5998	0,0333	3,4403	31,194	11,0009	10,009
Sistemos parametrai	$K=0,6; T=15 s; \tau=10 s;$											
a metodas	0,5967	0,55	15,1162	0,7747	8	20	0,602	0,3333	15,75	5	8	20
b metodas	0,5788	3,5333	11,1526	25,6493	11,5482	15,482	0,5791	3,4833	11,1266	25,8227	11,4950	14,95
Sistemos parametrai	$K=0,6; T=45 s; \tau=10 s;$											
a metodas	0,5740	4,3333	42,9363	4,586	7,3	27	0,5779	3,6833	43,1755	4,0544	7,2	28
b metodas	0,4027	32,8833	14,0430	68,7933	15,8066	58,066	0,403	32,8333	14,021	68,8422	15,7475	57,475

2.8 lentelė. Tyrimo metu gauti rezultatai

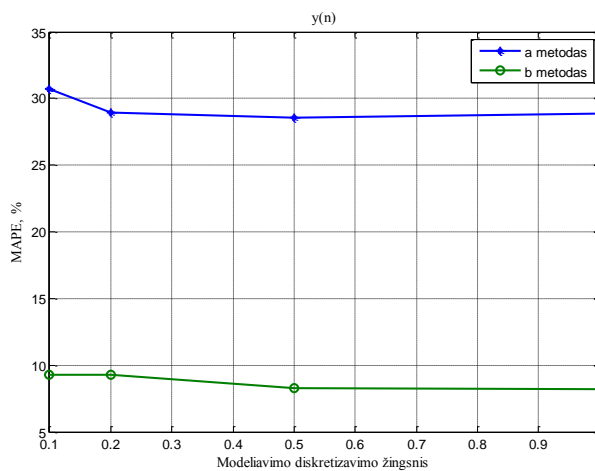
Duomenų nuskaitymo intervalas	$\Delta t=0,5s$						$\Delta t=1$					
Identifikuojami parametrai	K		T		τ		K		T		τ	
Parametras	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau=10 s;$											
a metodas	0,5999	0,0167	5,1196	2,392	9,5	5	0,6002	0,0333	4,8369	3,262	10	0
b metodas	0,6011	0,1833	4,082	18,36	10,2324	2,324	0,599	0,1667	2,0566	58,868	11,6642	16,642
Sistemos parametrai	$K=0,6; T1=15 s; \tau=10 s;$											
a metodas	0,6032	0,5333	15,8785	5,8567	8	20	0,6075	1,25	16,0184	6,7893	8	20
b metodas	0,5814	3,1	11,3794	24,1373	11,1735	11,735	0,5815	3,0833	11,2183	25,2113	10,9160	9,16
Sistemos parametrai	$K=0,6; T1=45 s; \tau=10 s;$											
a metodas	0,58	3,3333	43,3058	3,7649	7,5	25	0,6402	6,7	50,4267	12,0593	8	20
b metodas	0,4054	32,4333	14,209	68,4244	15,5317	55,317	0,4057	32,3833	14,0904	68,688	15,2402	52,402



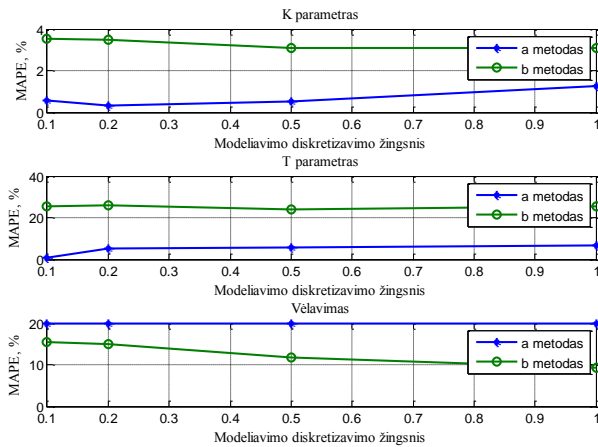
2.38 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



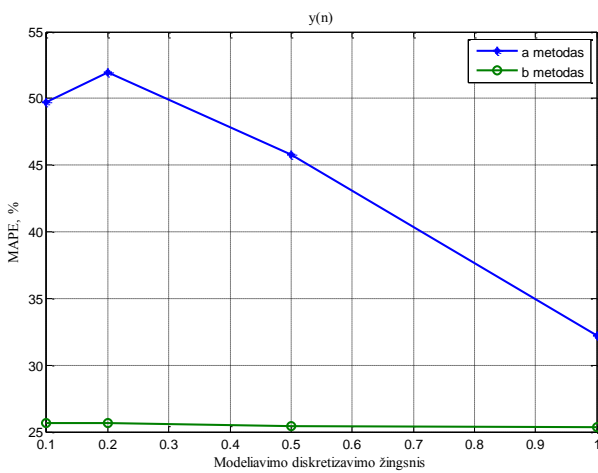
2.39 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 5s$



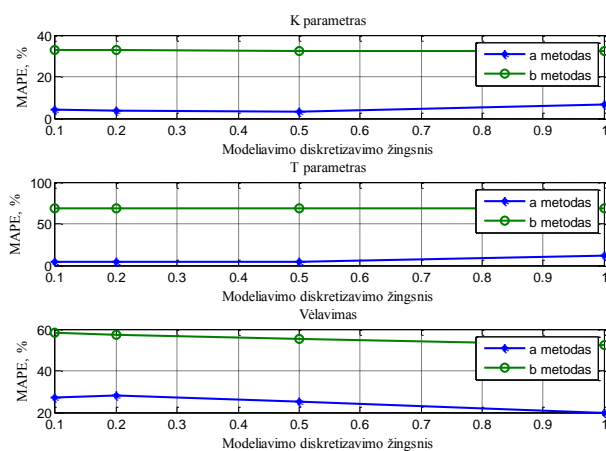
2.40 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.41 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 15s$



2.42 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



2.43 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 45s$

Atsiradus triukšmui, dingsta tiesinė priklausomybė tarp diskretizavimo žingsnio ir vidutinių absoliutinių parametų paklaidų. Atsiranda lūžio taškai, kai į abi puses keičiant diskretizavimo žingsnį identifikavimo kokybė blogėja.

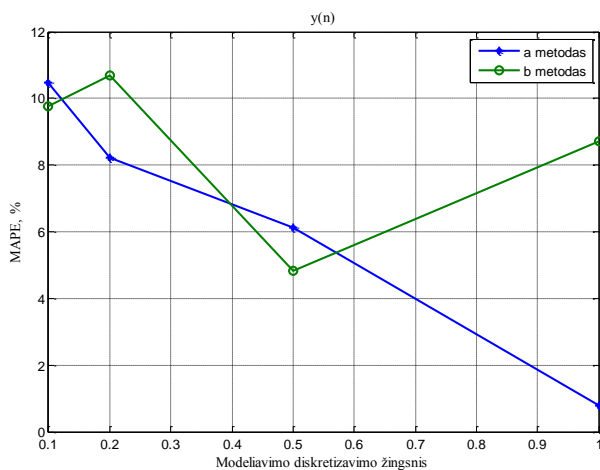
2.2.3 Sistemos modelio identifikavimas, kai signalas su 5% triukšmu

2.9 lentelė. Tyrimo metu gauti rezultatai

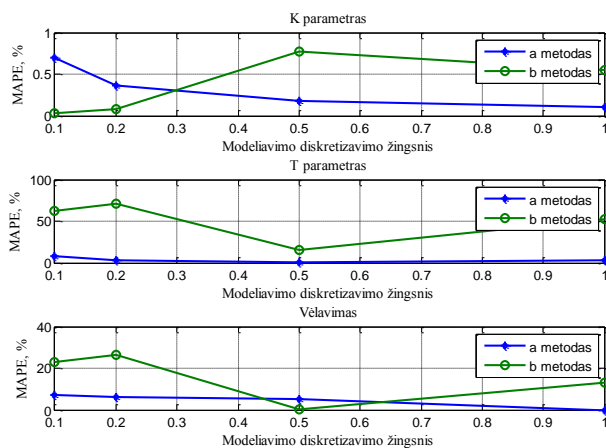
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$						$\Delta t=0,2s$					
Identifikuojami parametrai	K		T		τ		K		T		τ	
Parametras	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau=10 s;$											
a metodas	0,5958	0,7	4,6082	7,836	9,3	7	0,5978	0,3667	4,8199	3,602	9,4	6
b metodas	0,6002	0,0333	1,8930	62,14	12,3038	23,038	0,6005	0,0833	1,4464	71,072	12,6628	26,628
Sistemos parametrai	$K=0,6; T=15 s; \tau=10 s;$											
a metodas	0,5826	2,9	13,7565	8,29	8	20	0,5955	0,75	15,2414	1,6093	8	20
b metodas	0,5792	3,4667	10,9500	27	11,3886	13,886	0,5798	3,3667	10,9211	27,1927	11,3467	13,467
Sistemos parametrai	$K=0,6; T=45 s; \tau=10 s;$											
a metodas	0,4904	18,2667	32,4521	27,8842	7,30	27	0,5164	13,9333	35,3762	21,3862	7,4	26
b metodas	0,4031	32,8167	14,0750	68,7222	15,2480	52,48	0,4037	32,7167	14,0432	68,7929	15,2046	52,046

2.10 lentelė. Tyrimo metu gauti rezultatai

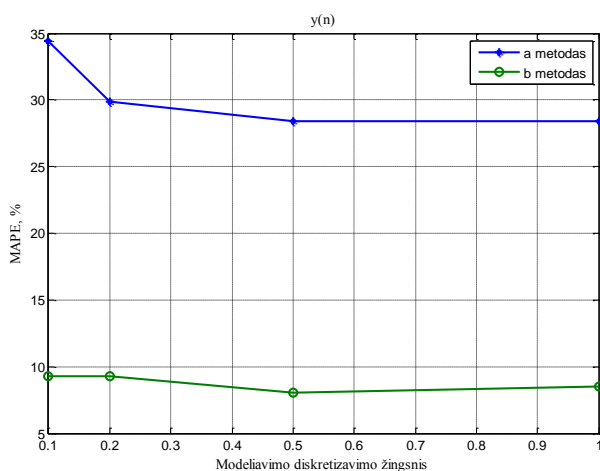
Duomenų nuskaitymo intervalas	$\Delta t=0,5s$						$\Delta t=1$					
Identifikuojami parametrai	K		T		τ		K		T		τ	
Parametras	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau=10 s;$											
a metodas	0,5989	0,1833	5,022	0,44	9,5	5	0,6006	0,1	4,8299	3,402	10	0
b metodas	0,6046	0,7667	4,2478	15,044	9,9621	0,379	0,6033	0,55	2,3682	52,636	11,2858	12,858
Sistemos parametrai	$K=0,6; T1=15 s; \tau=10 s;$											
a metodas	0,6005	0,0833	15,7922	5,2813	8	20	0,6103	1,7167	16,2433	8,2887	8	20
b metodas	0,5849	2,5167	11,5072	23,2853	10,8885	8,885	0,5859	2,35	11,3154	24,564	10,6879	6,879
Sistemos parametrai	$K=0,6; T1=45 s; \tau=10 s;$											
a metodas	0,5372	10,4667	37,9728	15,616	7,5	25	0,663	10,5	53,1973	18,2162	8	20
b metodas	0,4089	31,85	14,46	67,8667	14,9944	49,944	0,41	31,6667	14,2862	68,2529	14,7810	47,81



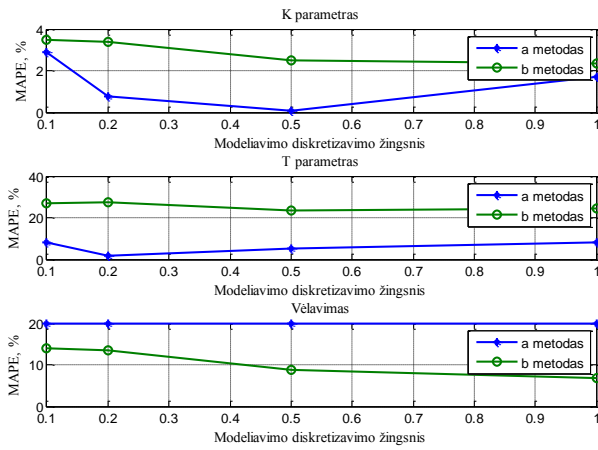
2.44 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



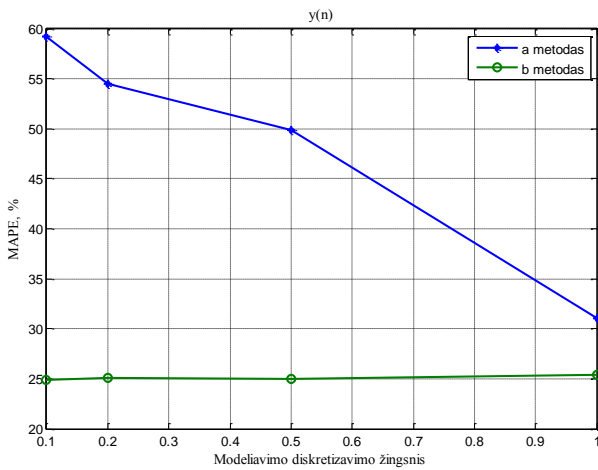
2.45 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 5s$



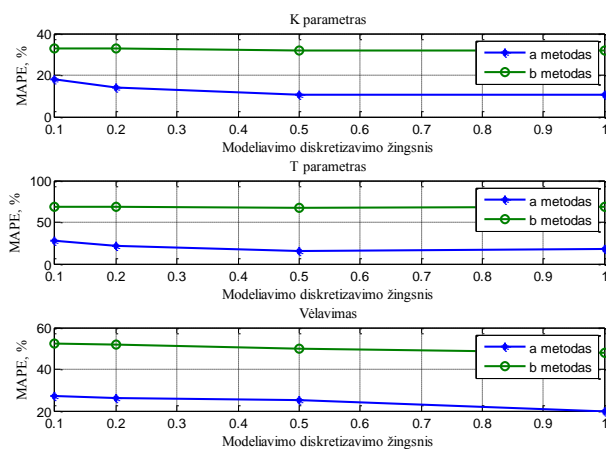
2.46 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.47 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 15s$



2.48 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



2.49 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 45s$

Iš atlikto eksperimento duomenų ir grafikų matyti, kad didinant triukšmą geresni identifikavimo rezultatai gaunami su didesniu diskretizavimo žingsniu.

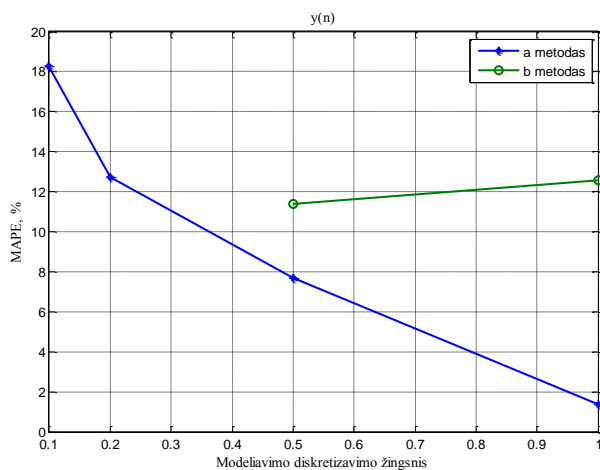
2.2.4 Sistemos modelio identifikavimas, kai signalas su 10% triukšmu

2.11 lentelė. Tyrimo metu gauti rezultatai

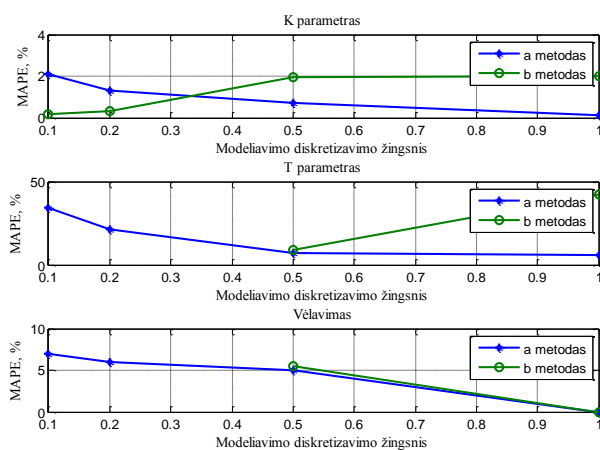
Duomenų nuskaitymo intervalas	$\Delta t=0,1s$						$\Delta t=0,2s$					
	K		T		τ		K		T		τ	
Identifikuojami parametrai	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau=10 s;$											
a metodas	0,5873	2,1167	3,2895	34,21	9,3	7	0,5921	1,3167	3,9305	21,39	9,4	6
b metodas	0,6009	0,15	-	-	-	-	0,6019	0,3167	-	-	-	-
Sistemos parametrai	$K=0,6; T1=15 s; \tau=10 s;$											
a metodas	0,5484	8,6	10,3935	30,71	8	20	0,5740	4,3333	13,3211	11,1927	8	20
b metodas	0,5798	3,3667	10,4965	30,0233	11,1189	11,189	0,5812	3,1333	12,4638	16,908	11,0990	10,99
Sistemos parametrai	$K=0,6; T1=45 s; \tau=10 s;$											
a metodas	0,3654	39,1	16,4283	63,4927	7,3	27	0,4103	31,6167	21,8437	51,4584	7,4	26
b metodas	0,4037	32,7167	14,0799	68,7113	14,1926	41,926	0,4051	32,4833	14,03	68,8222	14,1820	41,82

2.12 lentelė. Tyrimo metu gauti rezultatai

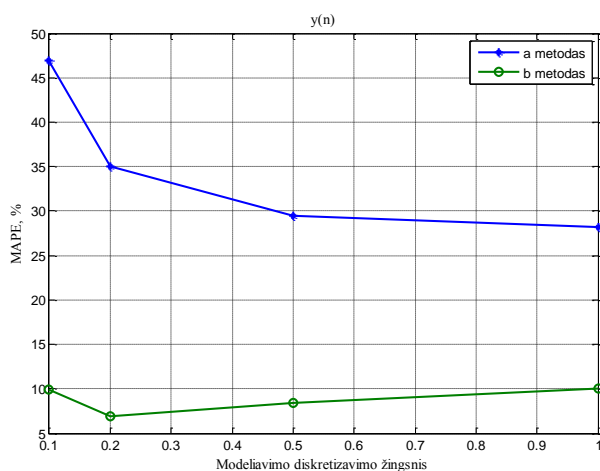
Duomenų nuskaitymo intervalas	$\Delta t=0,5s$						$\Delta t=1$					
Identifikuojami parametrai	K		T		τ		K		T		τ	
Parametras	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$	K	$\delta, \%$	T	$\delta, \%$	τ	$\delta, \%$
Sistemos parametrai	$K=0,6; T=5 s; \tau=10 s;$											
a metodas	0,5958	0,7	4,6413	7,174	9,5	5	0,6008	0,1333	4,7043	5,914	10	0
b metodas	0,6116	1,9333	4,5490	9,02	9,4555	5,445	0,6120	2	2,8777	42,446	10	0
Sistemos parametrai	$K=0,6; T1=15 s; \tau=10 s;$											
a metodas	0,5903	1,6167	15,1237	0,8247	8	20	0,6133	2,2167	16,4346	9,564	8	20
b metodas	0,5919	1,35	11,7481	21,6793	10,3386	3,386	0,5945	0,9167	11,4986	23,3427	10,2483	2,483
Sistemos parametrai	$K=0,6; T1=45 s; \tau=10 s;$											
a metodas	0,4557	24,05	27,6883	38,4704	7,5	25	0,6749	12,4833	54,4329	20,962	8	20
b metodas	0,4159	30,6833	14,9212	66,8418	13,9748	39,748	0,4187	30,2167	14,6446	67,4564	13,9122	39,122



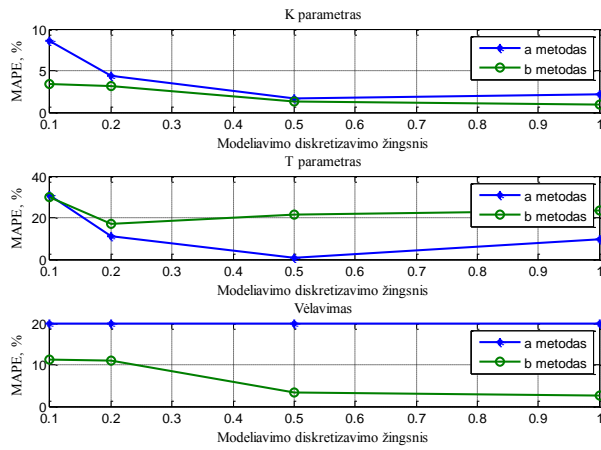
2.50 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 5s$



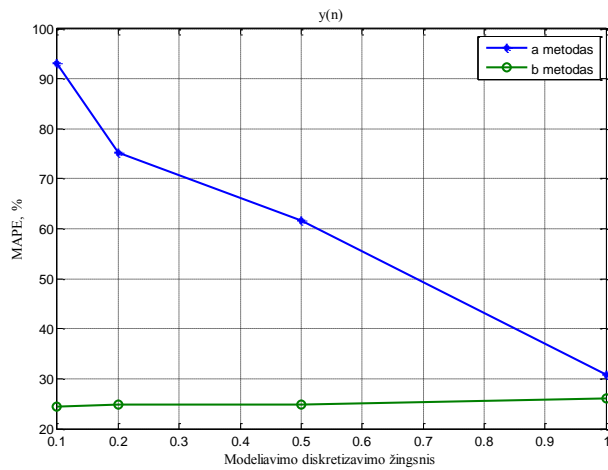
2.51 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametų vidutinėms absoliutinėms paklaidoms, kai $T = 5s$



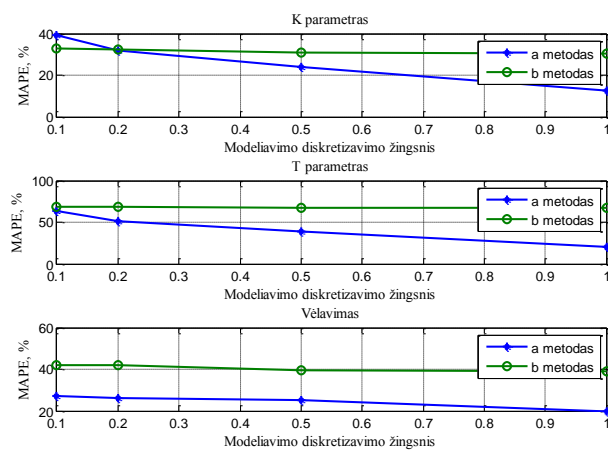
2.52 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 15s$



2.53 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 15s$



2.54 pav. Modeliavimo diskretizavimo žingsnio įtaka proceso vidutinei absoliutinei $y(k)$ paklaidai, kai $T = 45s$



2.55 pav. Modeliavimo diskretizavimo žingsnio įtaka modelio parametru vidutinėms absoliutinėms paklaidoms, kai $T = 45s$

Iš gautų identifikavimo duomenų matyti, kad didėjant triukšmui b identifikavimo metodui reikia pasirinkti didesnę diskretizavimo žingsnį, nes parinkus per mažą negalima tinkamai identifikuoti laiko pastoviosios ir vėlavimo.

3 Procesio identifikavimas ir reguliavimas

Atliekant proceso identifikavimo eksperimentus pastebėta, kad pirmos eilės sistemą su ir be vėlavimo galima pakankamai tiksliai identifikuoti.

Šiame skyriuje, naudojant anksčiau naudotus metodus, bus atliekami eksperimentai su slenkančiu duomenų langu ir bus bandoma identifikuoti pirmos eilės sistemą su vėlavimu. Tokios sistemos identifikavimas leis nustatyti regulatoriaus parametrus, nes dauguma derinimo metodikų naudoja tik pirmos eilės sistemos su vėlavimu parametrus.

Regulatoriaus parametrus derinti naudojamas IAE derinimo metodas [8].

Taikant a metodą, vėlavimo identifikavimas nėra tikslus. Tolimesniems tyrimams su šiuo metodu daroma prielaida, kad vėlavimas yra žinomas iš technologinių parametrų. Kad būtų galima palyginti abu identifikavimo metodus, b metode taip pat priimama prielaida, kad vėlavimas gali būti iš anksto nustatomas analizuojant technologinį procesą.

Kad būtų galima įvertinti identifikavimo kokybę, naudojama išraiška:

$$\text{Par_san.} = \frac{\frac{\sum_{i=1}^n x}{n}}{\frac{\sum_{i=1}^n z}{n}} \quad (80)$$

x – identifikuoto parametro vertė n momentu;

z – tikro parametro vertė n momentu.

Reguliavimo kokybei įvertinti naudojama išraiška:

$$\text{IAE} = \sum_{i=1}^n \text{abs}(\text{nuost.}(n) - y(n)) \quad (81)$$

nuost.(n) – nuostato reikšmė n momentu;

y(n) – sistemos išėjimo reikšmė n momentu.

3.1 Proceso identifikavimas iš slenkančio duomenų lango

Naudojant a ir b identifikavimo metodus, atlikti identifikavimo eksperimentai su slenkančiu duomenų langu. Tolimesniuose poskyriuose pateikti eksperimento rezultatai.

Modelio sudarymui naudojamas nebe „Matlab“ aplinkos modelis, o išraiška (78), kuri aprašo pirmos eilės sistemą be vėlavimo skirtuminėmis lygtimis.

$$y(n+1)=y(n)+dt*(1/T*(K*u(n)-y(n))) \quad (78)$$

$y(n)$ – proceso išėjimas, $u(n)$ – reikšmė proceso įėjime, dt – modeliavimo diskretiškumas, T - laiko pastovioji, K – stiprinimo koeficientas ir $y(n+1)$ –kito modeliavimo žingsnio proceso išėjimo reikšmė.

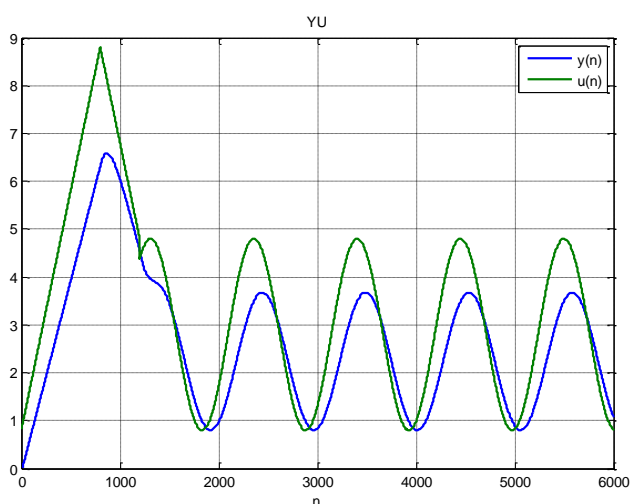
Ši išraiška gaunama iš diferencialinės lygties (79), kur $y(t)$ – proceso išėjimas, o $u(t)$ - įėjimas

$$(dy(t))/dt=1/T*(K*u(t)-y(t)) \quad (79)$$

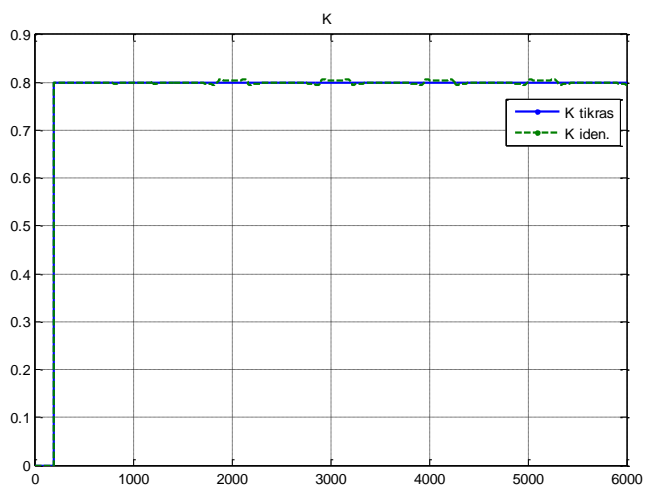
3.1.1 Identifikavimas, taikant diskretizuotas objekto įėjimo ir išėjimo reikšmes

„Matlab“ aplinkos kodas, skirtas pirmos eilės sistemos su vėlavimu parametru identifikavimui iš slenkančio duomenų lango, taikant diskretizuotas įėjimo ir išėjimo reikšmes, pateiktas 5 priede.

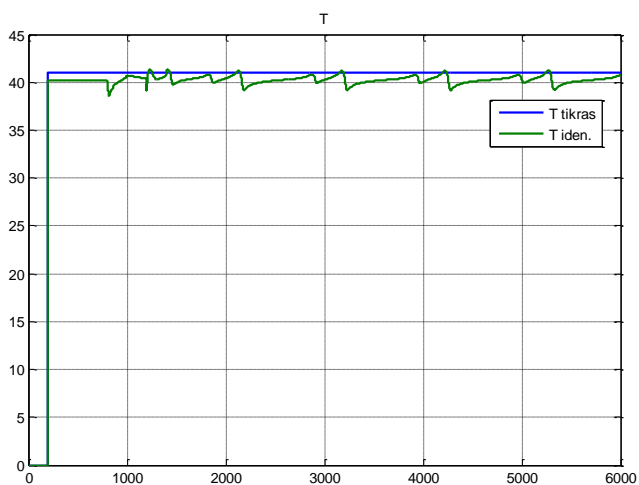
Parametru identifikavimas, kai K ir T pastovūs.



3.1 pav. Sistemos nuostato ir išėjimo duomenys grafiko pavidalu

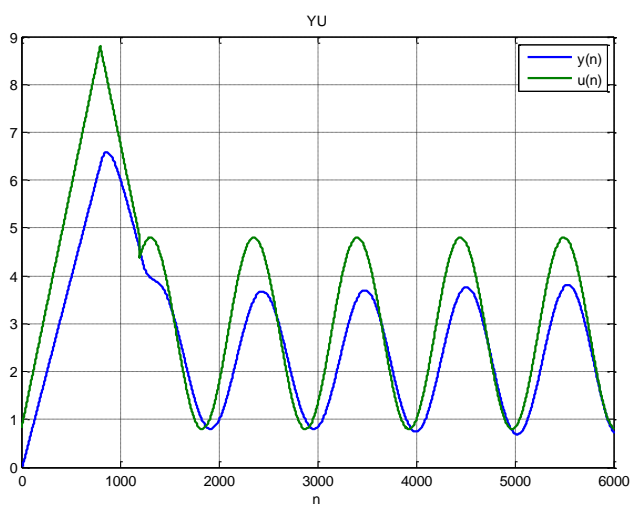


3.2 pav. Identifikuoto K parametro palyginimas su tikra reikšme

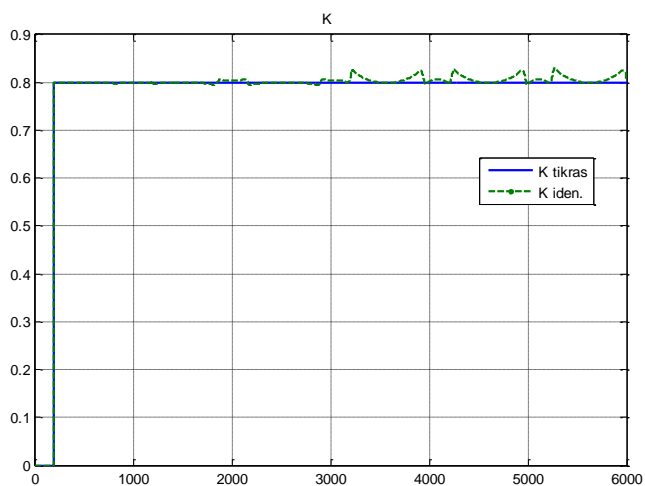


3.3 pav. Identifikuoto T parametro palyginimas su tikra reikšme

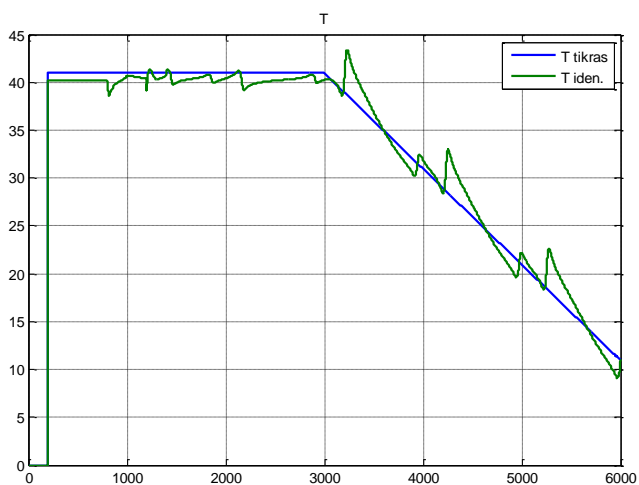
Parametrų identifikavimas, kai K pastovus, o T kinta.



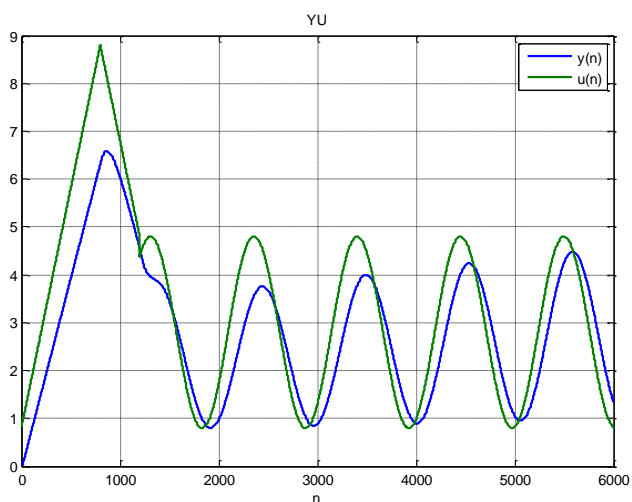
3.4 pav. Sistemos nuostato ir išėjimo duomenys grafiko pavidalu



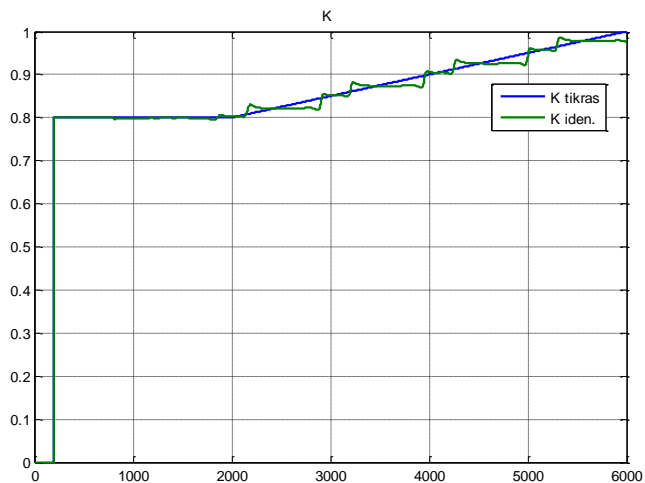
3.5 pav. Identifikuoto K parametro palyginimas su tikra reikšme



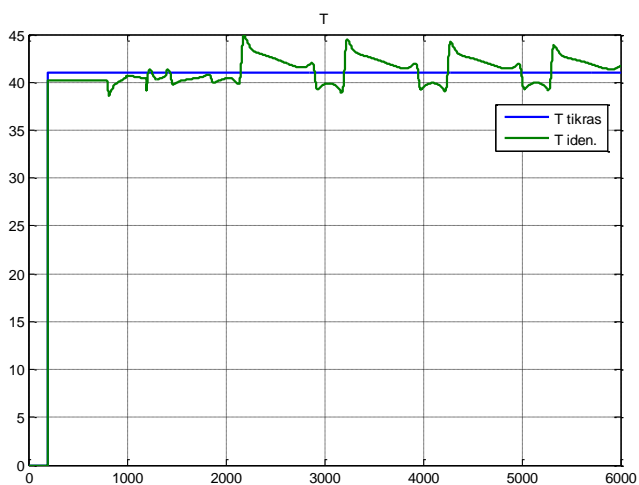
3.6 pav. Identifikuoto T parametro palyginimas su tikra reikšme
Parametrų identifikavimas, kai K kinta, o T pastovus.



3.7 pav. Sistemos nuostato ir išėjimo duomenys grafiko pavidalu

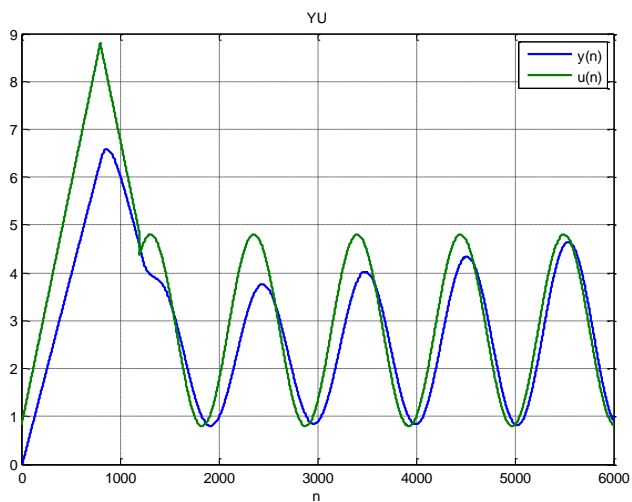


3.8 pav. Identifikuoto K parametro palyginimas su tikra reikšme

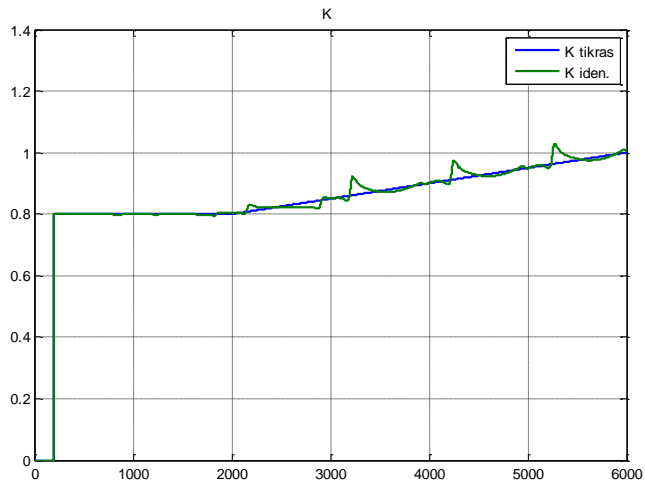


3.9 pav. Identifikuoto T parametro palyginimas su tikra reikšme

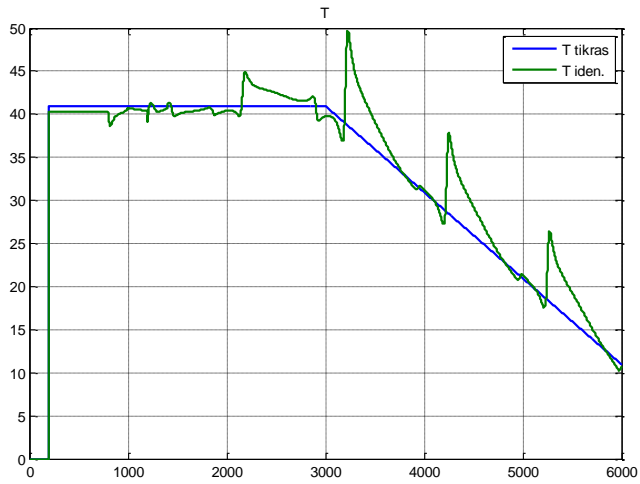
Parametrų identifikavimas, kai K ir T kinta.



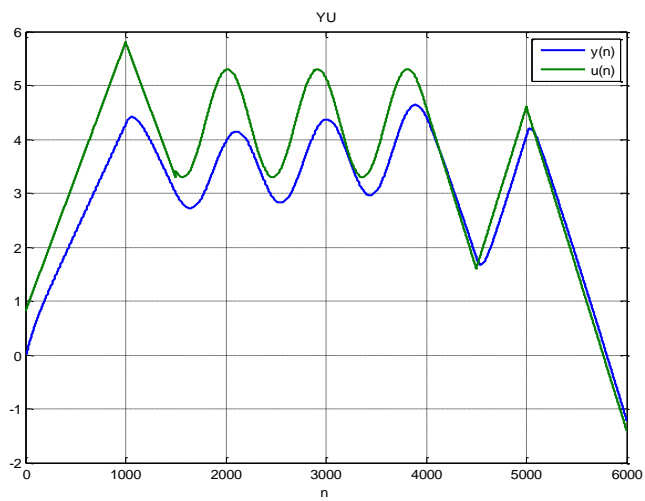
3.10 pav. Sistemos nuostato ir išėjimo duomenys grafiko pavidalu



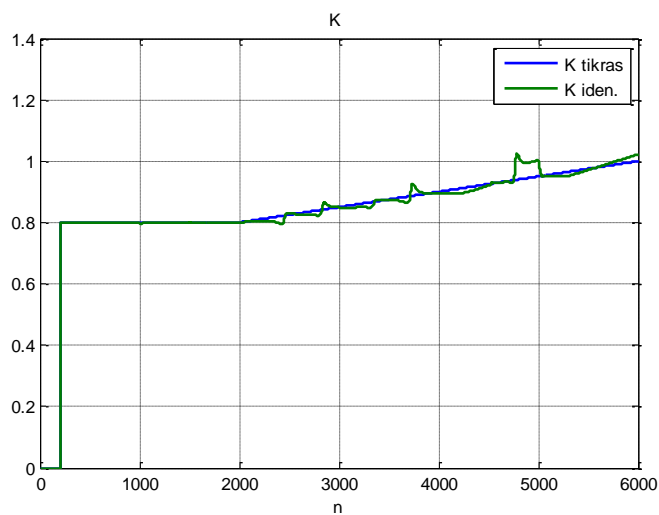
3.11 pav. Identifikuoto K parametro palyginimas su tikra reikšme



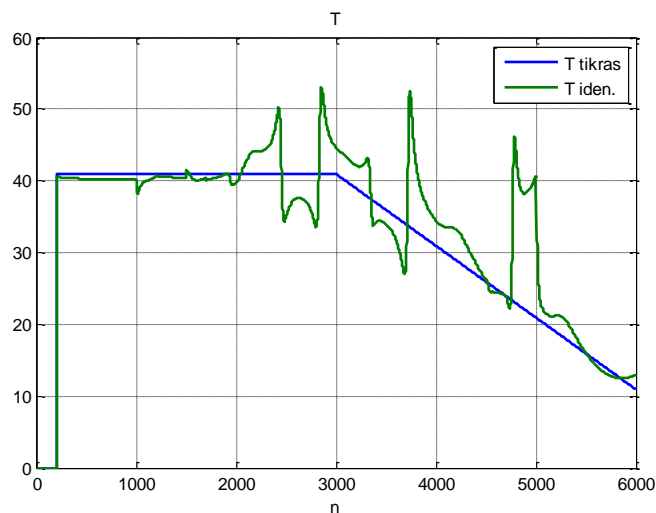
3.12 pav. Identifikuoto T parametro palyginimas su tikra reikšme



3.13 pav. Sistemos nuostato ir išėjimo duomenys grafiko pavidalu



3.14 pav. Identifikuoto K parametro palyginimas su tikra reikšme



3.15 pav. Identifikuoto T parametro palyginimas su tikra reikšme

3.1 lentelė. Identifikavimo santykiniai ir procentiniai įverčiai

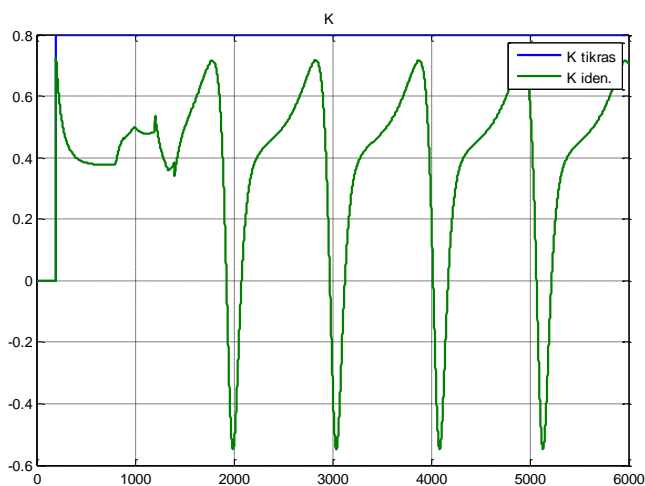
Parametras	Eksperimentas				
	K ir T pastovūs	K pastovus, T kinta	K kinta, T pastovus	K ir T kinta (4.10 paveikslas)	K ir T kinta (4.13 paveikslas)
K _{sant.}	0,9998	1,0044	0,9974	1,0025	1,0003
T _{sant.}	0,9821	0,9974	1,056	1,0266	1,0447
K δ , %	0,0238	0,4393	0,2621	0,2479	0,0327
T δ , %	1,7858	0,2603	0,5612	2,6614	4,4680

Iš atliktų eksperimentų matyti, kad naudojant a metodą parametų identifikavimas vyksta gana tiksliai ir išlieka parametų kitimo tendencija.

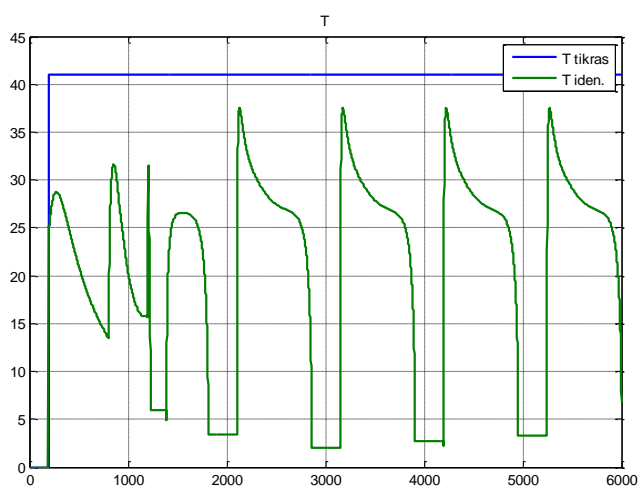
3.1.2 Identifikavimas, taikant netiesioginiu būdu suformuotas objekto impulsines reakcijos kreives

„Matlab“ aplinkos kodas, skirtas pirmos eilės sistemos su vėlavimu parametų identifikavimui iš slenkančio duomenų lango, taikant dirbtinai suformuotas impulsines reakcijos kreives, pateiktas 6 priede.

Parametų identifikavimas, kai K ir T pastovūs. Sistemos nuostato ir išėjimo duomenų grafikas pateiktas 4.1 paveiksle.



3.16 pav. Identifikuoto K parametro palyginimas su tikra reikšme



3.17 pav. Identifikuoto T parametro palyginimas su tikra reikšme

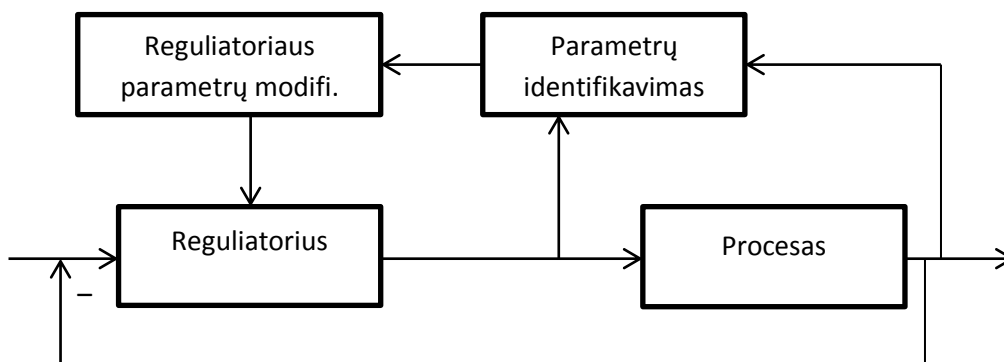
3.2 lentelė. Identifikavimo santykiniai įverčiai

Parametras	Eksperimentas
	K ir T pastovūs
K_sant.	0,4955
T_sant.	0,5104
K δ , %	50,4457
T δ , %	48,9586

Iš identifikuotų duomenų matyti, kad yra didelės paklaidos tarp tikro ir identifikuoto parametro, tad šis metodas netinka modelio parametrų identifikavimui, kai duomenų langas slenka, ir toliau šis metodas nebus naudojamas eksperimentams.

3.2 Proceso identifikavimas ir reguliavimas, naudojant slenkantį duomenų langą

Slenkančio duomenų lango pritaikymas procesui identifikuoti galėtų pagerinti proceso reguliavimo kokybę. Tad šiame skyriuje bus atliekami eksperimentai su kintančiais pagal identifikuotą procesą PI parametrais ir pastoviais PI parametrais, nustatytais pagal pradinis identifikuoto proceso parametrus. Algoritmo veikimo struktūrinė schema pavaizduota 4.18 paveiksle.



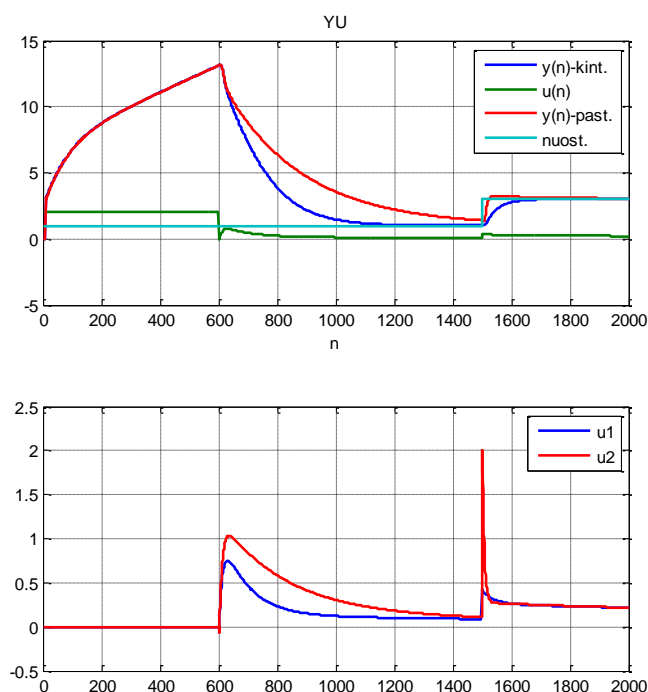
3.18 pav. Identifikavimo ir reguliavimo algoritmo struktūra

3.2.1 Identifikavimas ir reguliavimas, taikant diskretizuotas objekto įėjimo ir išėjimo reikšmes

„Matlab“ aplinkos kodas, skirtas pirmos eilės sistemos su vėlavimu parametru identifikavimui ir proceso reguliavimui, naudojant slenkantį duomenų langą, taikant diskretizuotas įėjimo ir išėjimo reikšmes, pateiktas 7 priede.

Atliktų eksperimentų grafikai:

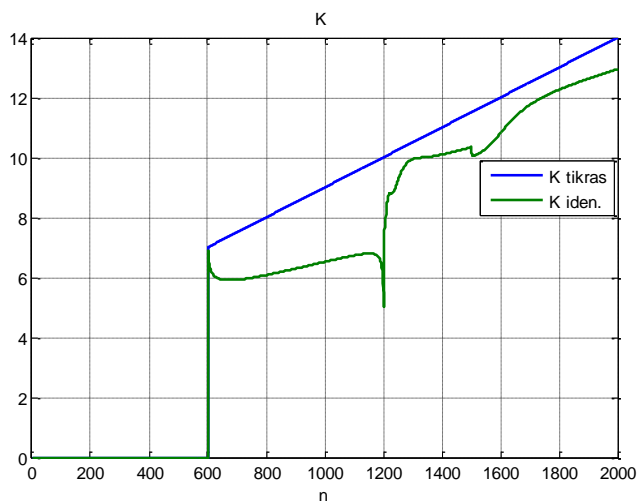
Stiprinimo koeficientas K parinktas kaip tiesinė funkcija, o laiko pastovioji T pastovi. Parinktas reguliatorius PI derinamas pagal nuostatą.



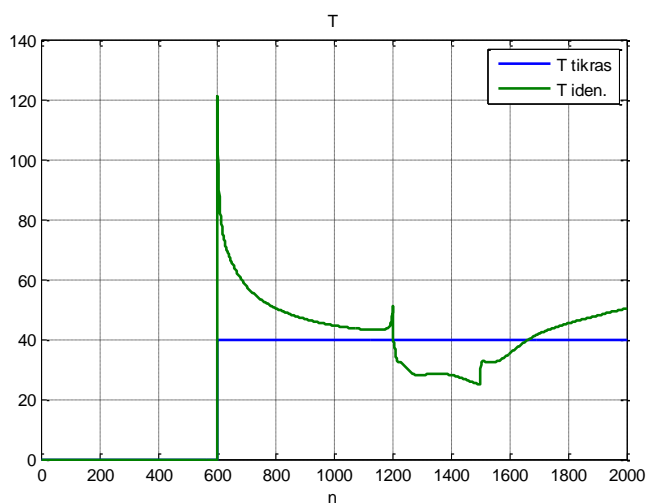
3.19 pav. Sistemos nuostatas, reguliatorių išėjimai ir sistemos išėjimas su pastoviais ir kintamais PI parametrais

Reguliatorių kokybės rodikliai:

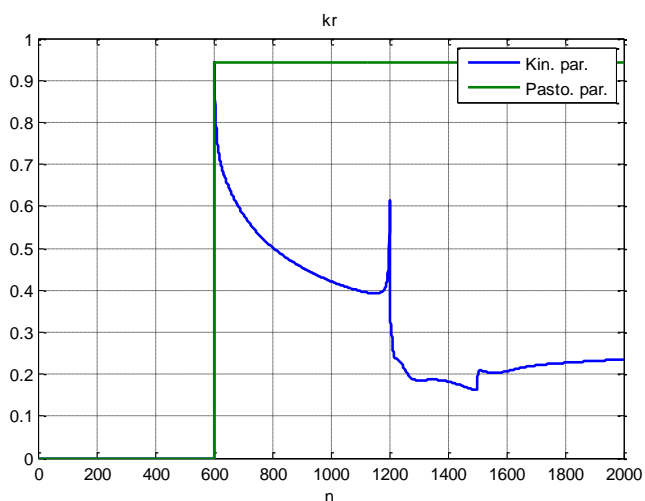
- su kintamais PI parametrais $IAE = 801,0379$;
- su pastoviais PI parametrais $IAE = 2006,4$.



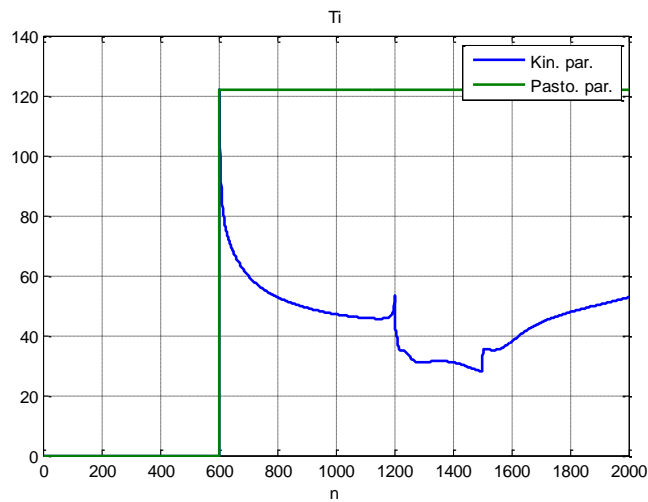
3.20 pav. Identifikuoto K parametro palyginimas su tikra reikšme



3.21 pav. Identifikuoto T parametro palyginimas su tikra reikšme

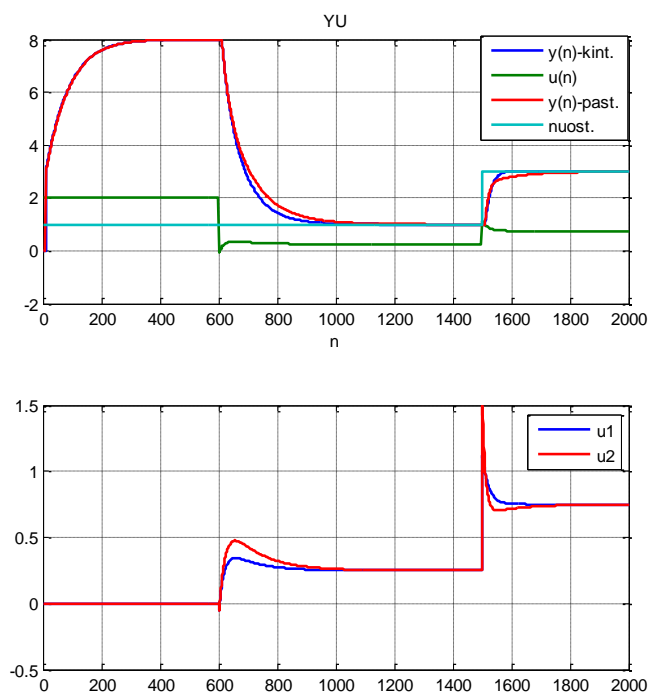


3.22 pav. Apskaičiuoti PI regulatoriaus pastovus ir kintamas kr parametrai, kai regulatorius derinamas pagal nuostatą



3.23 pav. Apskaičiuoti PI reguliatoriaus pastovus ir kintamas T_i parametrai, kai reguliatorius derinamas pagal nuostatą

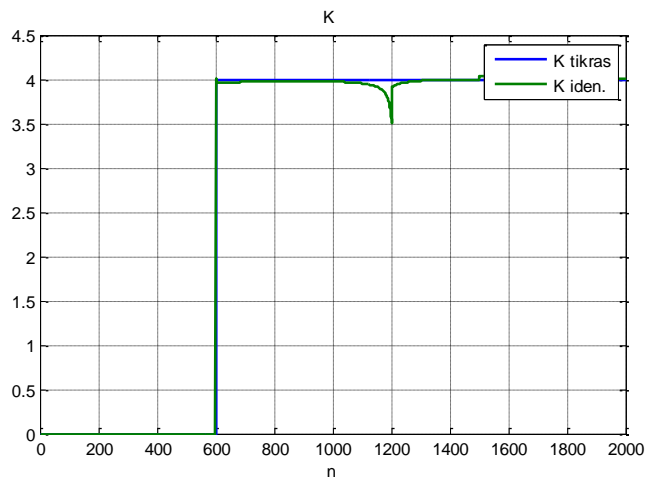
Stiprinimo koeficientas K parinktas pastovus, o laiko pastovioji T kaip tiesinė funkcija. Parinktas reguliatorius PI derinamas pagal nuostatą.



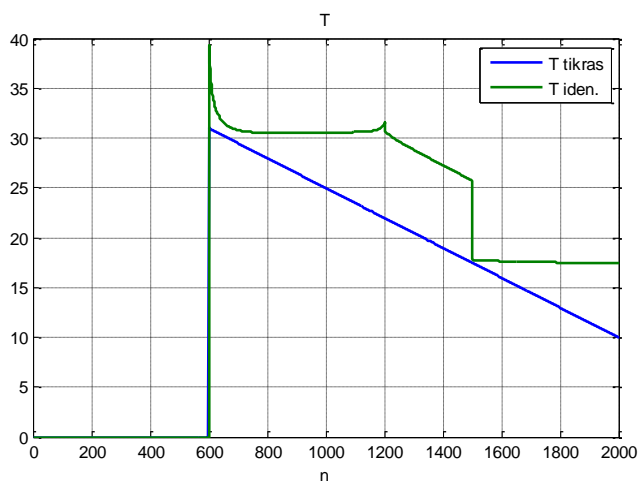
3.24 pav. Sistemos nuostatas, reguliatorių išėjimai ir sistemos išėjimas su pastoviais ir kintamais PI parametrais

Reguliatorių kokybės rodikliai:

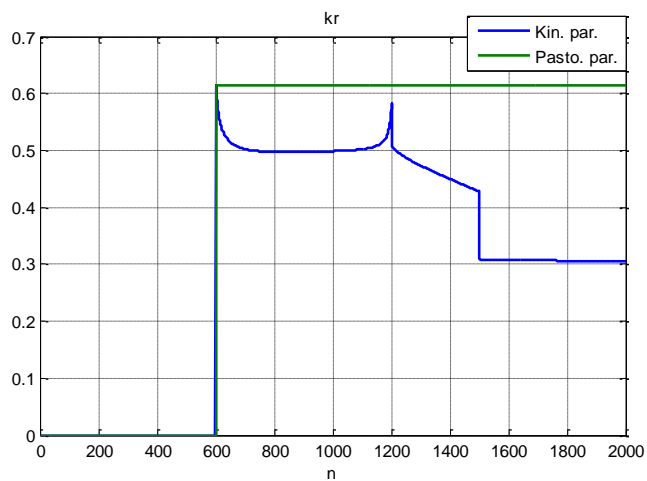
- su kintamais PI parametrais $IAE = 153,1107$;
- su pastoviais PI parametrais $IAE = 254,3858$.



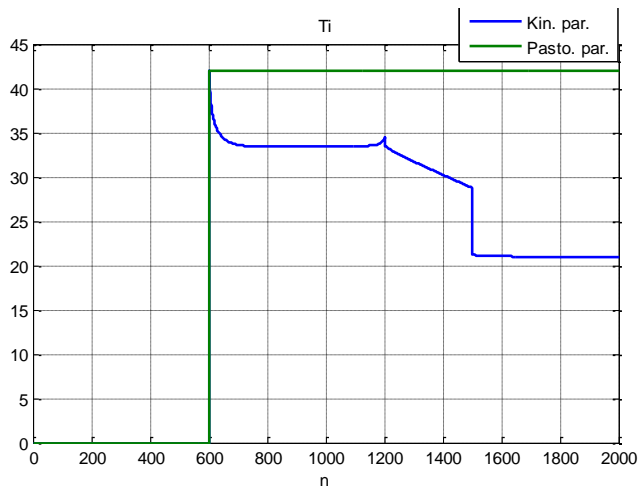
3.25 pav. Identifikuoto K parametro palyginimas su tikra reikšme



3.26 pav. Identifikuoto T parametro palyginimas su tikra reikšme

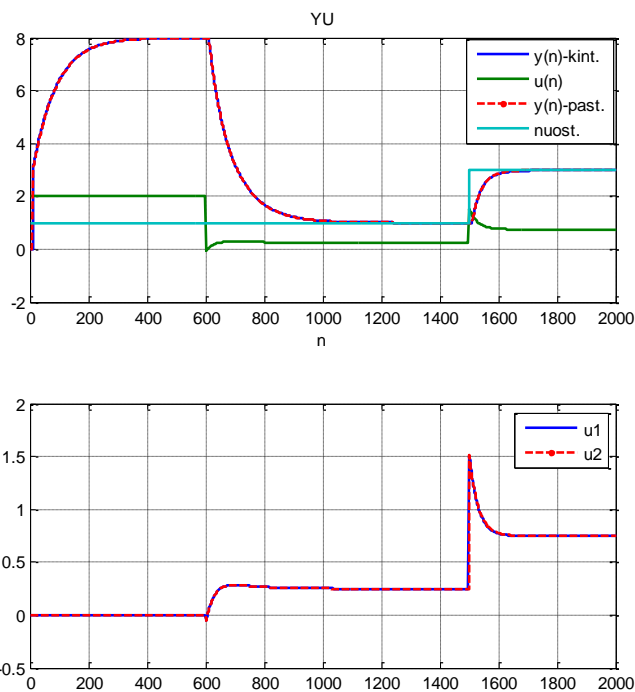


3.27 pav. Apskaičiuoti PI regulatoriaus pastovus ir kintamas kr parametrai, kai regulatorius derinamas pagal nuostatą



3.28 pav. Apskaičiuoti PI regulatoriaus pastovus ir kintamas T_i parametrai, kai reguliatorius derinamas pagal nuostatą

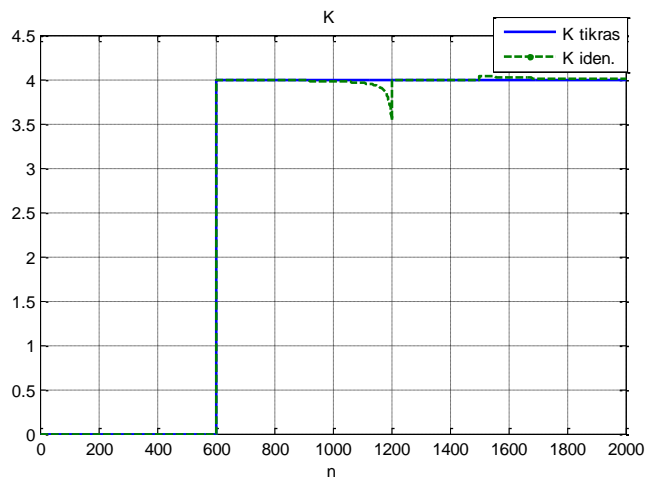
Stiprinimo koeficientas K ir laiko pastovioji T parinktos pastovios reikšmės. Parinktas reguliatorius PI derinamas pagal nuostatą.



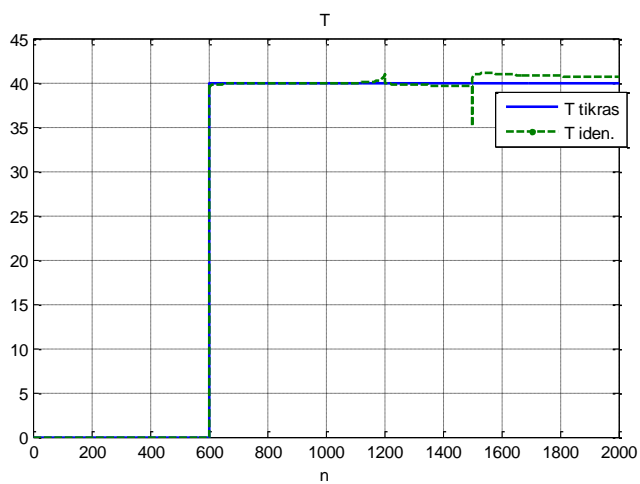
3.29 pav. Sistemos nuostatas, reguliatorių išėjimai ir sistemos išėjimas su pastoviais ir kintamais PI parametrais

Reguliatorių kokybės rodikliai:

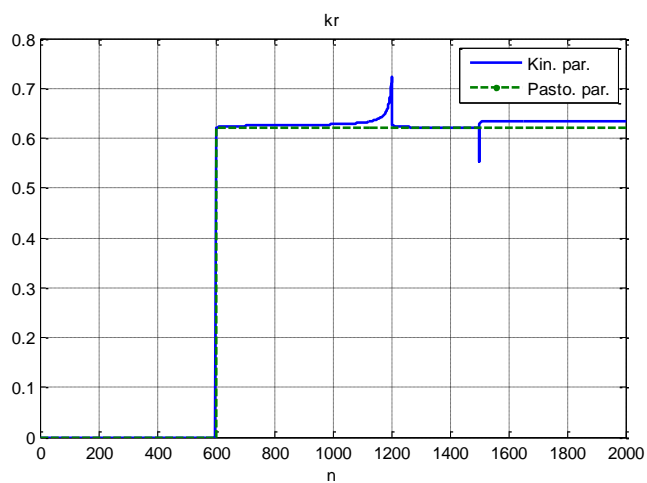
- su kintamais PI parametrais $IAE = 245,2250$;
- su pastoviais PI parametrais $IAE = 241,0353$.



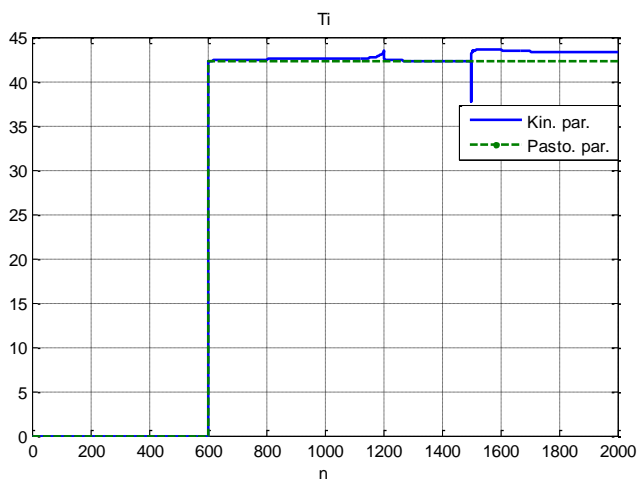
3.30 pav. Identifikuoto K parametro palyginimas su tikra reikšme



3.31 pav. Identifikuoto T parametro palyginimas su tikra reikšme

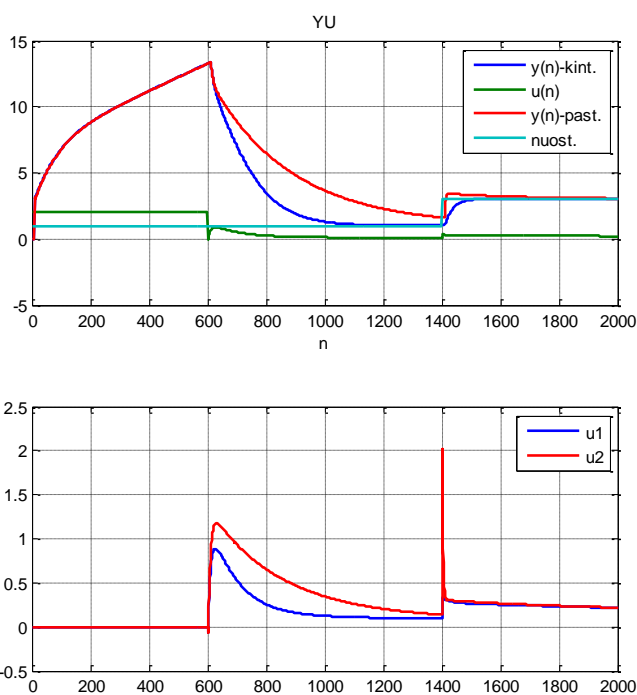


3.32 pav. Apskaičiuoti PI regulatoriaus pastovus ir kintamas k_r parametrai, kai regulatorius derinamas pagal nuostatą



3.33 pav. Apskaičiuoti PI regulatoriaus pastovus ir kintamas T_i parametrai, kai reguliatorius derinamas pagal nuostatą

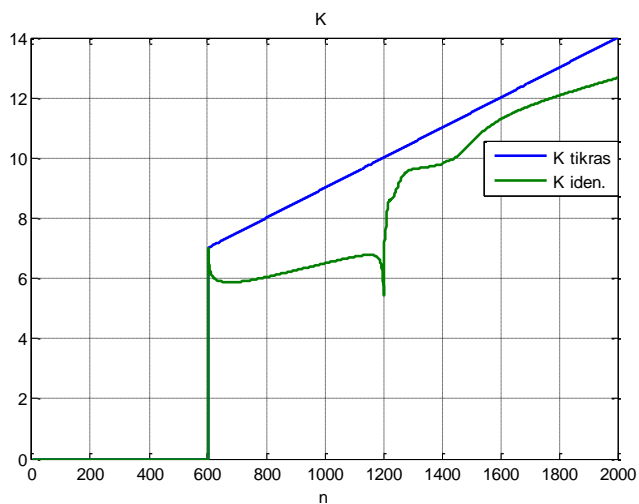
Stiprinimo koeficientas K ir laiko pastovioji T parinktos kaip tiesinės funkcijos. Parinktas reguliatorius PI derinamas pagal nuostatą.



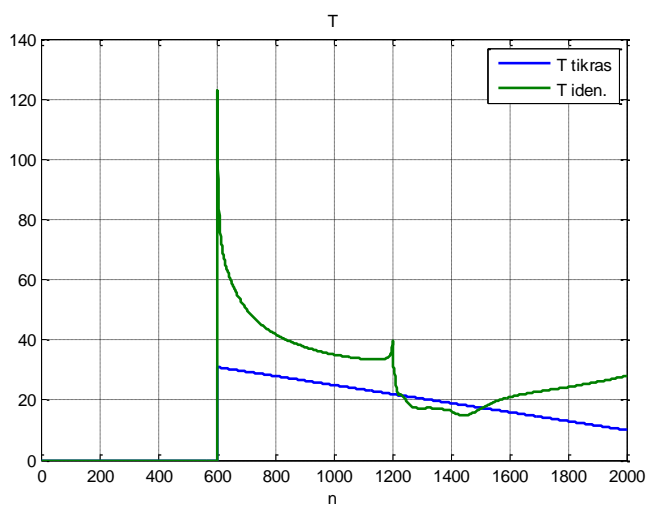
3.34 pav. Sistemos nuostatas, reguliatorių išėjimai ir sistemos išėjimas su pastoviais ir kintamais PI parametrais

Reguliatorių kokybės rodikliai:

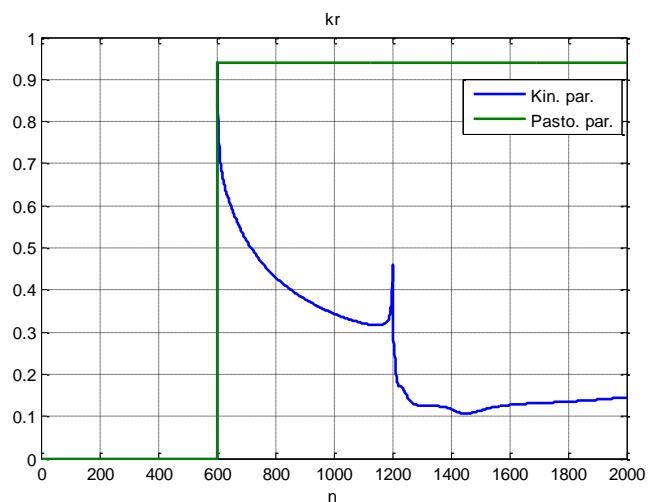
- su kintamais PI parametrais $IAE = 671,2056$;
- su pastoviais PI parametrais $IAE = 2066,6$.



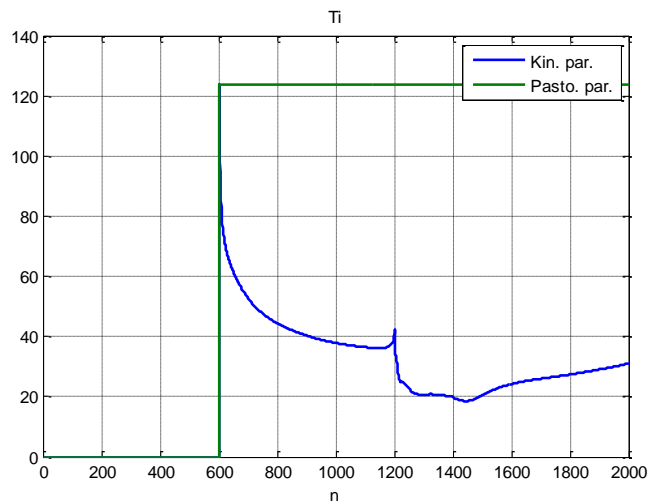
3.35 pav. Identifikuoto K parametro palyginimas su tikra reikšme



3.36 pav. Identifikuoto T parametro palyginimas su tikra reikšme



3.37 pav. Apskaičiuoti PI regulatoriaus pastovus ir kintamas kr parametrai, kai regulatorius derinamas pagal nuostatą



3.38 pav. Apskaičiuoti PI regulatoriaus pastovus ir kintamas T_i parametrai, kai regulatorius derinamas pagal nuostatą

3.3 lentelė. Identifikavimo ir reguliavimo kokybės įverčiai

Parametras	Eksperimentas			
	K kinta, T pastovus	K pastovus, T kinta	K ir T pastovūs	K ir T kinta
K _{sant.}	0,8587	0,9959	0,9980	0,8540
T _{sant.}	1,0756	1,2458	1,0052	1,4751
K δ , %	14,1292	0,4109	0,1988	14,6015
T δ , %	7,5641	24,5831	0,5150	47,5050
PID _{kok_kin.}	801,0379	153,1107	245,2250	671,2056
PID _{kok_pas.}	2006,4	254,3858	241,0353	2066,6

Atliekant eksperimentus pastebėta, kad kai modelio parametrai kinta, geresnis reguliavimo kokybės įvertinimas gaunamas su kintančiais regulatoriaus parametrais.

4 Rezultatai ir išvados

1. Atlikus literatūros apžvalgą, pasirinkti lengvai realizuojami metodai procesų modelių identifikavimui, kurie suprogramuoti „Matlab“ aplinkoje. Imitacinio modeliavimo eksperimentai parodė, kad juos galima taikyti nesudėtingų procesų identifikavimui, panaudojant proceso normalaus funkcionavimo duomenis.
2. Atliktas eksperimentinis tyrimas parodė, kad naudojant normalaus funkcionavimo duomenų diskretinius duomenų matavimus ir dirbtinai iš jų suformavus impulsines reakcijos kreives, galima tinkamai identifikuoti pirmos eilės sistemas be vėlavimo ir su vėlavimu parametrus (identifikavimo kokybe buvo patikima, vidutinė absoliutinė procentinė paklaida nuo 0,1 % iki 5 % tikslumu, parinkus tinkama diskretiškumą).
3. Kai modelio parametrai keičiasi proceso metu, taikant slenkanti duomenų langą ir naudojant metodą su diskretinių momentų duomenimis, rezultatai buvo pakankamai tikslūs (identifikuotų parametru vidutinė absoliutinė procentinė paklaida neviršijo 4,5 %, kai nėra matavimo triukšmų). Šis metodas rekomenduojamas taikyti procesams, kai modelio parametrai keičiasi.
4. Taikant slenkanti duomenų langą, metodas paremtas dirbtinai suformuota impulsine reakcijos kreive, buvo netikslus ir jo nerekomenduojama taikyti, kai modelio parametrai kinta.
5. Modelio parametru identifikavimo tikslumas uždaramoje kontūre buvo pakankamas, kad daugumoje atveju užtikrinti geresnę reguliavimo kokybę negu reguliuojant su pastoviais regulatoriaus parametrais (IAE pagerėjo nuo 40% iki 67 %, priklausomai nuo procesų įvedimo sąlygų).

5 Literatūros šaltiniai

1. Simutis R. Sistemų modeliavimas ir identifikavimas – „VPU“ leidykla 2012 – 177 psl.
2. A. Ogunnaike, W. H. Ray. Process dynamics, modeling and control. Oxford University Press, 1994, 1260p
3. A Real-Time Identification Method of a Slow Process Parameters Using Adaptive IIR-OSLMS Filters control [žiūrėta 2016 – 04 – 05] – Prieiga per Internetą:
<http://www.wseas.us/e-library/transactions/systems/2008/28-319.pdf>
4. Y.Zhu Multivariable system identification for process control – Elsevier Science & Technology Books 2001, 352p
5. T. Tekorius, V. Galvanauskas. Procesų ir sistemų modeliavimas. - Kaunas : Technologija, 2011.
6. A novel online affine model identification of multivariable processes using adaptive neuro-fuzzy networks [žiūrėta 2016 – 04 – 20] – Prieiga per Internetą
<http://www.sciencedirect.com/science/article/pii/S0263876209001816>.
7. Finddelay [žiūrėta 2016 – 05 – 01] – Prieiga per Internetą
<http://se.mathworks.com/help/signal/ref/finddelay.html>.
8. D. Levišauskas Automatinio reguliavimo sistemų derinimas – „VPU“ leidykla 2008 – 118 psl.
9. Lennart Ljung Identification for control: simple process models – Las Vegas 2002 – 9 psl.
10. K. J. _Aström and T. Hägglund. *PID Controllers: Theory, Design, and Tuning* - Instrument Society of America 1995 – 346 psl.

Priedai

1 priedas

Identifikavimas iš visų sukauptų duomenų, jei nėra vėlavimo, taikant diskretizuotas įėjimo signalo ir objekto reakcijos kreivės reikšmes

```
clear all
clc
load y;% užkraunami sistemos išėjimo duomenys
load u;% užkraunami sistemos įėjimo duomenys
del_t=1;%pasirenkamas diskretizavimo žingsnis
% Suformuojamos duomenų matricos
Xx(58,2)=0;
Yx(58,1)=0;
Xx(:,1)=y(2:59);
Xx(:,2)=u(2:59);
Yx(:,1)=y(3:60);
% Mažiausių kvadratų metodu randami parametrai a1 ir b1
a=(inv(Xx'*Xx))*Xx'*Yx;
a1=a(1,1);
b1=a(2,1);
% Apskaičiuojami parametrai laiko pastovioji ir stiprinimo koeficientas
tau=-del_t/log(a1)
K=b1/(1-a1)
```

2 priedas

Identifikavimas iš visų sukauptų duomenų, jei nėra vėlavimo, taikant netiesioginiu būdu suformuotas objekto impulsines reakcijos kreives.

```
clc
clear all
% Užkrauname eksperimentinius duomenis
load duomenys.txt;
[Nrow,Ncol]=size(duomenys);
for j=2:Nrow
    u(j-1)=duomenys(j-1,2);
    y(j-1)=duomenys(j,3);
    time(j-1)=duomenys(j-1,1);
end
% Jei diskretiškumas yra dt ne 1 , tai
% reikia įvesti santykini laika, kad diskretiškumas pasidarytu 1
% tam paskaičiuojame daugikli ts:
ts=1/(time(2)-time(1));
time_s=time*ts;
% Suformuojame matricą U
M=Nrow-1;
U=zeros(M);
for i=1:M
    k=0;
    for j=i:M
        k=k+1;
        U(j,i)=u(k);
    end
end
end
% Išskaičiuojame impulsinės reakcijos kreivės taškus g
g=inv(U)*y';
% Nustatome perdavimo funkcijos parametrus
% Suskaičiuojame momentus,
% integravimą atliekame, naudojant trapecijų metodą
```

```
m0=trapz(time_s,g);
m1=trapz(time_s,time_s.*g')/m0;
% Apskaičiuojami parametrai: laiko pastovioji ir stiprinimo koeficientas
K=m0
tau1_s=m1;
% Tikra laiko pastovioji gaunama padalinus iš santykio ts
tau1=tau1_s/ts
```

3 priedas

Identifikavimas iš visų sukauptų duomenų, jei yra vėlavimas, taikant diskretizuotas įėjimo signalo ir objekto reakcijos kreivės reikšmes

```
Clc
clear all
load y;% užkraunami sistemos išėjimo duomenys
load u;% užkraunami sistemos įėjimo duomenys
del_t=1; ;%pasirenkamas diskretizavimo žingsnis
% priklausomai nuo diskretizavimo žingsnio sudaromų matricių pirmas skaičius parenkamas taip,
kad būtų daugiau nei 0
Xx(:,1)=y(11:59);
Xx(:,2)=u(11:59);
Yx(:,1)=y(12:60);
% Atliekamas parametrų normavimas
M1=mean(Xx(:,2));
S1=std(Xx(:,2));
NDu=(Xx(:,2)-M1)/S1
M2=mean(Xx(:,1));
S2=std(Xx(:,1));
NDy=(Xx(:,1)-M2)/S2
% Vėlavimo identifikavimas
Delay=finddelay(NDu,NDy)
% įvertinus vėlavimą keičiami duomenų intervalai
Xxx(:,1)=y((11+Delay):59);
Xxx(:,2)=u(11:(59-Delay));
Yxx(:,1)=y((12+Delay):60);
% Mažiausių kvadratų metodu randami parametrai a1 ir b1
a=(inv(Xxx'*Xxx))*Xxx'*Yxx;
a1=a(1,1);
b1=a(2,1);
% Apskaičiuojami parametrai: laiko pastovioji ir stiprinimo koeficientas
tau1=-del_t/log(a1)
K=b1/(1-a1)
```


4 priedas

Identifikavimas iš visų sukauptų duomenų, jei yra vėlavimas, taikant netiesioginiu būdu suformuotas objekto impulsines reakcijos kreives.

```
clc
clear all
% Užkrauname eksperimentinius duomenis
load duomenys.txt;
[Nrow,Ncol]=size(duomenys);
for j=2:Nrow
    u(j-1)=duomenys(j-1,2);
    y(j-1)=duomenys(j,3);
    time(j-1)=duomenys(j-1,1);
end
% Jei diskretiškumas yra dt ne 1 , tai
% reikia įvesti santykini laika, kad diskretiškumas pasidarytu 1
% tam paskaičiuojame daugikli ts:
ts=1/(time(2)-time(1));
time_s=time*ts;
% Suformuojame matricą U
M=Nrow-1;
U=zeros(M);
for i=1:M
    k=0;
    for j=i:M
        k=k+1;
        U(j,i)=u(k);
    end
end
end
g=inv(U)*y';% Formuojama impulsinė reakcijos kreivė
% Nustatome perdavimo funkcijos parametrus
% Suskaičiuojame momentus,
% integravimą atliekame, naudojant trapecijų metodą
m0=trapz(time_s,g);
```

```

m1=trapz(time_s,time_s.*g')/m0;
m2=trapz(time_s,(time_s.^2.*g'))/m0;
% Apskaičiuojami parametrai: laiko pastovioji, vėlavimas ir stiprinimo koeficientas
K=m0
vel_t_1_s=(2*m1+sqrt(((2*m1)^2)-4*(2*(m1^2)-m2)))/2
vel_t_2_s=(2*m1-sqrt(((2*m1)^2)-4*(2*(m1^2)-m2)))/2
tau1_1s=m1-vel_t_1_s;
tau1_2s=m1-vel_t_2_s;
% Tikri laiko pastovioji ir vėlavimas gaunami, padalinus iš santykio ts
tau1=tau1_1s/ts
tau2=tau1_2s/ts
vel_1=vel_t_1_s/ts
vel_2=vel_t_2_s/ts

```

5 priedas

Identifikavimas iš slenkančio duomenų lango, jei vėlavimas žinomas, taikant diskretizuotas įėjimo signalo ir objekto reakcijos kreivės reikšmes

```
clc, clear all
dt=0.5;
yy(1)=0;
u(1)=1;
KK=0.8;
tau=41;
delay=10;
T=200;
for n=1:6000
    % Laiko pastoviosios kitimas
    if n<3000
        tau=41;
    else tau=41-(n-3000)*0.01;
    end
    % Stiprinimo koef. kitimas
    if n<2000
        %KK=n*0.0001
        KK=0.8;
    else KK=0.75+(n-1000)*0.00005;
    end
    % Duodamas į sistemos įėjimą šuolis
    if n<800
        u(n)=0.8+n*0.01;
    else if n<1200
        u(n)=0.8+800*0.01-(n-800)*0.01 ;
    else
        u(n)=2.8+sin(n*0.006)*2;
    end
end
% Formuojamas sistemos išėjimas
```

```

yy(n+1)=yy(n)+dt*((1/tau)*(KK*u(n)-yy(n)));
yy(n)=yy(n+1);
% Duomenys išsaugomi matricoje
YU(n+delay,1)=yy(n);
YU(n,2)=u(n);
YUU(n,1)=yy(n);
YUU(n,2)=u(n);
    if n>T
        Xx(:,1)=YU(n-T:n-1,1);
        Xx(:,2)=YU(n-T:n-1,2);
        Yx(:,1)=YU(n+1-T:n,1);
        % Atliekamas parametru normavimas vėlavimo identifikavimui
        M1=mean(Xx(:,2));
        S1=std(Xx(:,2));
        NDU=(Xx(:,2)-M1)/S1;
        M2=mean(Xx(:,1));
        S2=std(Xx(:,1));
        NDY=(Xx(:,1)-M2)/S2;
        % Vėlavimo identifikavimas
        %Delay=sqrt((finddelay(NDU,NDY))^2);
        Delay=10;
        % įvertinus vėlavimą keičiami duomenų intervalai
        % vietoj 10 delay turētu būti
        Xxx(:,1)=YUU(n-T:n-1,1);
        Xxx(:,2)=YUU(n-T:n-1,2);
        Yxx(:,1)=YUU(n+1-T:n,1);
        % Mažiausių kvadratų metodu randami parametrai a1 ir b1
        aa=(inv(Xxx'*Xxx))*Xxx'*Yxx ;
        a1=aa(1,1);
        b1=aa(2,1);
        % Apskaičiuojami parametrai: laiko pastovioji ir stiprinimo koeficientas
        tau1=-dt/log(a1);
        K=b1/(1-a1);
        % Duomenys išsaugomi matricoje
        Par(n,1)=K;

```

```

Par(n,2)=tau1;
Par(n,3)=Delay ;
% Duomenys išsaugomi matricose, kad būtų galima palyginti
ParK(n,1)=KK;
ParK(n,2)=Par(n,1);

ParT(n,1)=tau;
ParT(n,2)=Par(n,2);
    end
end

%Parametru identifikavimo kokybės įvertinimas
K_sant=mean(ParK(T:n,2))/mean(ParK(T:n,1))
T_sant=mean(ParT(T:n,2))/mean(ParT(T:n,1))
K_MAPE=abs((mean(ParK(T:n,1))-mean(ParK(T:n,2)))/mean(ParK(T:n,1)))*100
T_MAPE=abs((mean(ParT(T:n,1))-mean(ParT(T:n,2)))/mean(ParT(T:n,1)))*100

figure(1)
plot(YU(1:n,:)), grid on
title('YU')
legend('y(n)', 'u(n)')
xlabel ('n')
figure (2)
plot(ParK(1:n,:)), grid on
title('K ')
legend('K tikras', 'K iden.')
figure (3)
plot(ParT(1:n,:)), grid on
title('T')
legend('T tikras', 'T iden.')

```

6 priedas

Identifikavimas iš slenkančio duomenų lango, jei vėlavimas žinomas, taikant netiesioginiu būdu suformuotas objekto impulsines reakcijos kreives.

```
clc, clear all
dt=0.5;
yy(1)=0;
u(1)=1;
KK=0.8;
tau=41;
delay=0;
T=200;
for n=1:6000
    % Laiko pastoviosios kitimas
    %if n<3000
    tau=41;
    %else tau=41-(n-3000)*0.01;
    %end
    % Stiprinimo koef. kitimas
    % if n<2000
    %KK=n*0.0001
    KK=0.8;
    % else KK=0.75+(n-1000)*0.00005;
    %end
    % Duodamas į sistemos įėjimą šuolis
    if n<800
    u(n)=0.8+n*0.01;
    else if n<1200
        u(n)=0.8+800*0.01-(n-800)*0.01 ;
    else if n<6000
        u(n)=2.8+sin(n*0.006)*2;
    else if n<7000
        u(n)=0.8+(n-6000)*0.005;
    else u(n)=5.8;
```

```

        end
    end
end
end
% Formuojamas sistemos išėjimas
yy(n+1)=yy(n)+dt*((1/tau)*(KK*u(n)-yy(n)));
yy(n)=yy(n+1);
% Duomenys išsaugomi matricoje
data(n,1)=n*dt;
data(n+delay,3)=yy(n);
data(n,2)=u(n);
if n>T
    duomenys=[];
    duomenys = data(n-T:n-1,:);
    [Nrow,Ncol]=size(duomenys);
    for j=2:Nrow
        uid(j-1)=duomenys(j-1,2);
        y(j-1)=duomenys(j,3);
        time(j-1)=duomenys(j-1,1);
    end
% Jei diskretiškumas yra dt ne 1 , tai
% reikia įvesti santykini laika, kad diskretiškumas pasidarytu 1
% tam paskaičiuojame daugikli ts:
ts=1/(time(2)-time(1));
time_s=time*ts;
% Suformuojame matricą U
M=Nrow-1;
U=zeros(M);
for i=1:M
    k=0;
    for j=i:M
        k=k+1;
        U(j,i)=uid(k);
    end
end
end

```

```

g=inv(U)*y';% Formuojama impulsinė reakcijos kreivė
% Nustatome perdavimo funkcijos parametrus
% Suskaičiuojame momentus,
% integravimą atliekame, naudojant trapecijų metodą
m0=trapz(time_s,g);
m1=trapz(time_s,time_s.*g')/m0;
m2=trapz(time_s,(time_s.^2.*g')/m0);
K=m0;
%%%%%%%%%%%%
vel_t_1_s=(2*m1+sqrt(((2*m1)^2)-4*(2*(m1^2)-m2)))/2;
vel_t_2_s=(2*m1-sqrt(((2*m1)^2)-4*(2*(m1^2)-m2)))/2;
tau1_1s=m1-vel_t_1_s;
tau1_2s=m1-vel_t_2_s;
% Gauta laiko pastoviąją reikia padalyti iš santykio ts
tau1=tau1_1s/ts;
tau2=tau1_2s/ts;
vel_1=vel_t_1_s/ts;
vel_2=vel_t_2_s/ts;
% Duomenys išsaugomi matricoje
Par(n,1)=K;
Par(n,2)=tau2;
Par(n,3)=0 ;
% Įvedami parametų apribojimai, kad skaičiavimų paklaidos nepaveiktu
% rezultatų
if abs(imag(Par(n,2)))>0
Par(n,2)=Par(n-1,2);
end
% Duomenys išsaugomi matricose, kad būtų galima palyginti
ParK(n,1)=KK;
ParK(n,2)=Par(n,1);

ParT(n,1)=tau;
ParT(n,2)=Par(n,2);

ParV(n,1)=delay;

```



```

ParV(n,2)=Par(n,3);

end

end

%Parametrų identifikavimo kokybės įvertinimas
K_sant=mean(ParK(T:n,2))/mean(ParK(T:n,1))
T_sant=mean(ParT(T:n,2))/mean(ParT(T:n,1))
K_MAPE=abs((mean(ParK(T:n,1))-mean(ParK(T:n,2)))/mean(ParK(T:n,1)))*100
T_MAPE=abs((mean(ParT(T:n,1))-mean(ParT(T:n,2)))/mean(ParT(T:n,1)))*100

figure(1)
plot(data(1:n,2:3)), grid on
title('YU')
legend('y(n)', 'u(n)')
figure (2)
plot(ParK(1:n,:)), grid on
title('K ')
legend('K tikras', 'K iden.')
figure (3)
plot(ParT(1:n,:)), grid on
title('T')
legend('T tikras', 'T iden.')
%figure (4)
%plot(ParV(1:n,:)), grid on
%title('Delay')
%legend('delay tikras', 'delay iden.')

```

7 priedas

Identifikavimas ir reguliavimas (uždaro kontūro reguliavimo sistema) iš slenkančio duomenų lango, jei vėlavimas žinomas, taikant diskretizuotas įėjimo signalo ir objekto reakcijos kreivės reikšmes

```
warning off
clc, clear all
dt=0.5;%Diskretizavimo žingsnis
yy(1)=3;
yyy(1)=3;
KK=3;%Stiprinimo koef.
ttau=10;%Laiko pastovioji
delay=10;%Vėlavimas
T=600;%Slenkančio lango ilgis
u1(1)=0;
u2(1)=0;
cc=0;
for n=1:2000
    % Stiprinimo koef. kitimas
    KK=4+n*0.005;
    % Duodamas į sistemos įėjimą šuolis
    if n<1400
        sp(n)=1;
    else sp(n)=3;
    end
    % Laiko pastoviosios kitimas
    ttau=40-n*0.015;
    % Formuojamas sistemos išėjimas
    if n<T+2
        s_p(n)=2;
        u1(n)=0;
        u2(n)=0;
        yy(n+1)=yy(n)+dt*((1/ttau)*(KK*s_p(n)-yy(n)));
        yy(n)=yy(n+1);
    end
end
```

```

yyy(n+1)=yy(n);
else
% Reguliavimas su kintamais PID parametrais
u_delta(n)=(kr*((1+Td/dt+dt/Ti)*e(n-1)-(1+2*Td/dt)*e(n-2)+Td/dt*e(n-3)));
u1(n)=u_delta(n)+u1(n-1);
yy(n+1)=yy(n)+dt*((1/tau)*(KK*u1(n)-yy(n)));
yy(n)=yy(n+1);
s_p(n)=u1(n);
% Reguliavimas su pastoviais PID parametrais
uu_delta(n)=(krr*((1+Tdd/dt+dt/Tii)*ee(n-1)-(1+2*Tdd/dt)*ee(n-2)+Tdd/dt*ee(n-3)));
u2(n)=uu_delta(n)+u2(n-1);
yyy(n+1)=yyy(n)+dt*((1/tau)*(KK*u2(n)-yyy(n)));
yyy(n)=yyy(n+1);
end
e(n)= sp(n)-yy(n);
ee(n)= sp(n)-yyy(n);

% Duomenys išsaugomi matricoje
YU(n+delay,1)=yy(n);
YU(n,2)=s_p(n);
YU(n+delay,3)=yyy(n);
YU(n,5)=u1(n);
YU(n,6)=u2(n);
YU(n,4)=sp(n);
YU(n,7)=yy(n);

if abs(imag(YU(n+delay,1)))>0
YU(n+delay,1)=YU(n+delay-1,1);
end

if n>T
Delay=10;
% įvertinus vėlavimą keičiami duomenų intervalai
% vietoj 10 delay turētu būti
Xx(:,1)=YU(n-T:n-1,7);

```

```

Xx(:,2)=YU(n-T:n-1,2);
Yx(:,1)=YU(n+1-T:n,7);
% Mažiausių kvadratų metodu randami parametrai a1 ir b1
aa=(inv(Xx'*Xx))*Xx'*Yx;
a1=aa(1,1);
b1=aa(2,1);
% Apskaičiuojami parametrai: laiko pastovioji ir stiprinimo koeficientas
tau1=-dt/log(a1);
K=b1/(1-a1);
% Duomenys išsaugomi matricoje
Par(n,1)=K;
Par(n,2)=tau1;
Par(n,3)=Delay ;
% Įvedami parametru apribojimai, kad skaičiavimų paklaidos nepaveiktu
% rezultatų
if abs(imag(Par(n,1)))>0
Par(n,1)=Par(n-1,1);
end
if abs(imag(Par(n,2)))>0
Par(n,2)=Par(n-1,2);
end
if isnan((Par(n,1)))
Par(n,:)=Par(n-1,:);
end

% Duomenys išsaugomi matricose, kad būtų galima palyginti
ParK(n,1)=KK;
ParK(n,2)=Par(n,1);
%%%%%%%%%%
ParT(n,1)=ttau;
ParT(n,2)=Par(n,2);
%%%%%%%%%%
ParV(n,1)=delay;
ParV(n,2)=Par(n,3);
% Regulatoriaus parametru skaičiavimas IAE metodu

```

```

k0=Par(n,1);
T0=Par(n,2);
tau0=Par(n,3);
CONTROLLER='PI'; % PI arba PID
INP='sp' ; % SP arba DIST
A=[...
%|-----DIST-----|-----SP-----|
%| PI
0.984 0.986 0.758 0.861;...%| kr
0.608 0.707 1.020 0.323;...%| Ti
%| PID-----|-----|
1.435 0.921 1.086 0.869;...%| kr
0.878 0.749 0.740 0.130;...%| Ti
0.482 1.137 0.348 0.914;...%| Td
%| -----|-----|
];
switch upper(INP)
    case 'DIST'
        col_shift=[1:2];
    case 'SP'
        col_shift=[3:4];
    otherwise
        disp ('Error defining input signal type!');
end
switch upper(CONTROLLER)
    case 'PI'
        raw_shift=[1:2];
    case 'PID'
        raw_shift=[3:5];
    otherwise
        disp ('Error defining controller type!');
end
% Pasirinktas PARAMETRU segmentas
a=A(raw_shift,col_shift);
kr= a(1,1)/(k0*(tau0/T0)^a(1,2));

```

```

switch upper(INP)
    case 'DIST'
        Ti= T0/a(2,1)*(tau0/T0)^a(2,2) ;
    case 'SP'
        Ti= T0/(a(2,1)-a(2,2)*(tau0/T0));
    otherwise
end
switch upper(CONTROLLER)
    case 'PI'
        Td= 0;
    case 'PID'
        Td= a(3,1)*(tau0/T0)^a(3,2)*T0;
    otherwise
end
PID(n,1)=kr;
PID(n,2)=Ti;
PID(n,3)=Td;
% Pirmi nustatyti PID parametrai naudojami, kaip pastovūs
if cc==0
while PID(n,1)>0
    cc=n;
    break
end
end
%%%%%%%%%%
krr=PID(cc:cc,1);
Tii=PID(cc:cc,2);
Tdd=PID(cc:cc,3);
% Duomenys išsaugomi matricose, kad būtų galima palyginti
KR(n,1)=PID(n,1);
KR(n,2)=krr;
%%%%%%%%%%
TI(n,1)=PID(n,2);
TI(n,2)=Tii;
%%%%%%%%%%

```

```

    TD(n,1)=PID(n,3);
    TD(n,2)=Tdd;
    %%%%%%%%%%
    rer_kok1(n)=abs(sp(n)-yy(n));
    rer_kok2(n)=abs(sp(n)-yyy(n));
end
end
%Parametru identifikavimo kokybės įvertinimas
K_sant=mean(ParK(T:n,2))/mean(ParK(T:n,1))
T_sant=mean(ParT(T:n,2))/mean(ParT(T:n,1))

K_MAPE=abs((mean(ParK(T:n,1))-mean(ParK(T:n,2)))/mean(ParK(T:n,1)))*100
T_MAPE=abs((mean(ParT(T:n,1))-mean(ParT(T:n,2)))/mean(ParT(T:n,1)))*100

%Reguliavimo kokybės įvertinimas
PID_1=sum(rer_kok1(700:n))
PID_2=sum(rer_kok2(700:n))
%%%%%%%%%
figure(1)
subplot(2,1,1)
plot(YU(1:n,1:4)), grid on
title('YU')
legend('y(n)-kint.','u(n)','y(n)-past.','nuost.')
xlabel('n')
subplot(2,1,2)
plot(YU(1:n,5:6)), grid on
legend('u1','u2')
figure (2)
plot(ParK(1:n,:)), grid on
title('K ')
legend('K tikras','K iden.')
xlabel('n')
figure (3)
plot(ParT(1:n,:)), grid on
title('T')

```

```
legend('T tikras','T iden.')
xlabel ('n')
figure (5)
plot(KR(1:n,:)), grid on
title('kr')
legend('Kin. par.','Pasto. par.')
xlabel ('n')
figure (6)
plot(TI(1:n,:)), grid on
title('Ti')
legend('Kin. par.','Pasto. par.')
xlabel ('n')
```