

Informacinės sistemos išvystymas elektros energijai taupyti išmaniajame būste

Dalė Dzemydienė

Vilniaus universiteto Duomenų mokslo ir skaitmeninių technologijų instituto profesorė, vyr. mokslo darbuotoja
Vilnius University Institute of Data Science and Digital Technologies, Professor, scientific researcher
Akademijos g. 4, LT-08663 Vilnius, Lietuva
El. paštas: dale.dzemydiene@mii.vu.lt

Evaldas Žulkas

Vilniaus universiteto doktorantas,
Klaipėdos universiteto lektorius
Vilnius University lecturer,
Klaipėda University doctoral student
El. paštas: zulkas.e@gmail.com

Santrauka. *Informacinės sistemos atlieka svarbų vaidmenį valdant išmanųjį būstą. Jose kaupiami įrenginių, jungiamų prie belaidžio tinklo (pvz., jutiklių, šviestuvų, termometrų) stebėsenos duomenys. Tokio tipo sistemos priskiriamos kompleksinėms šiuolaikinėms elektroninių paslaugų valdymo sistemoms, galinčioms suteikti aplinkos valdymo paslaugas išmaniajame būste. Tačiau, kad išmaniajame būste veikiančios prietaisai galėtų efektyviai naudoti elektros energijos išteklius, reikia papildomų programinių modulių, leidžiančių tuos įrenginius optimaliai naudoti. Mūsų mokslinio tiriamojo darbo tikslas – išvystyti informacinės sistemos funkcijas, kad būtų galima prognozuoti elektros sąnaudas išmaniojo būsto sistemoje ir prognozavimo duomenis panaudoti valdant šias energijos sąnaudas.*

Reikšminiai žodžiai: *informacinė sistema, daiktų interneto technologija, išmanaus būsto valdymo paslaugos, elektros energijos valdymo sistema*

Išmaniojo būsto sistemos, kurios integruoja elektros sąnaudų valdymo funkcijas ir leidžia išmaniai valdyti prie belaidžio interneto prijungtus įvairius prietaisus, priskiriamos daiktų interneto technologiniams sprendimams. Dažniausiai išmaniojo būsto valdymo sistemose integruojamos apšvietimo, šildymo, vėdinimo posistemės, kurioms reikalingi specialūs prietaisai ir tinklų galimybės aplinkos parametrus nuskaityti. Parametru

stebėsenai ir duomenų surinkimui projektuojamos informacinės sistemos, kurių duomenys naudojami vertinant situacijas ir priimant sprendimus (Jiang X., 2010; Bielskis ir kt., 2009; Chen ir kt., 2008; Kaklauskas, Zavadskas ir kt., 2015).

Siekiant priimti automatizuotus sprendimus energijos sąnaudoms reguliuoti, tiriamos įterptinių sistemų duomenų surinkimo, perdavimo ir analizės posistemių galimybės

ir jų taikymas efektyvesniam energijos išteklių naudojimui. Darbe siūloma išvystyti elektros energijos sąnaudų valdymo sistemos architektūrą, sutelkiant dėmesį į energijos suvartojimo poreikius, sąnaudas ir jų galimus prognozės bei efektyvesnio naudojimo metodus. Duomenys perduodami tarp įrenginių naudojant telemetrijos technologiją ir mašinos bendravimo su mašina principą M2M (Adinya, Daoliang, 2012; Chen ir kt., 2008). Kuriant energijos sąnaudų valdymo (ESV) sistemą, mūsų keliamas uždavinys – pritaikyti autonominio elektros įrenginių sąnaudų valdymo principą ir į kuriamą ESV sistemą įtraukti prognozavimu grindžiamus sprendimo priėmimo metodus (Dzemydiene ir kt., 2015; Andziulis ir kt. 2009).

Literatūroje nagrinėjamos energijos sąnaudų valdymo sistemose valdymas nėra grindžiamas prognozavimo analize (Barnawi ir kt. 2016; Daniela ir kt. 2015; Engerati, 2016). Mūsų eksperimentinių tyrimų uždavinys – integruoti į energijos sąnaudų valdymo sistemą prognozavimo galimybes ir pagal prognozės pasirinktus scenarijus pasiūlyti korekcinius valdomų prietaisų režimus.

Sprendimų priėmimo posistemė vertina praėjusio laikotarpio informacinės sistemos duomenis ir pagal prognozuojamus elektros energiją taupančius scenarijus sumažina arba padidina įrenginių veikimo intensyvumą. Tokiais moduliais papildyta sistema duoda pagrindą sukurti gana adaptyvią, vartotojų poreikiams pritaikomą elektroninių paslaugų sistemą. Sistemos autonomiškumui užtikrinti keliami tam tikri reikalavimai, t. y. į sistemą įterpiami įrenginiai (jutikliai ir valdikliai) turėtų gauti vykdymo paskirtį realiu laiku ir ši paskirtis galėtų būti keičiama.

1. Trumpa išmaniojo būsto elektros energijos valdymo sistemų apžvalga

Analizuojant būsto (t. y. namų) energijos valdymo sistemas pastebima, kad viena iš pagrindinių užduočių yra efektyviai naudoti elektros energijos išteklius ir patalpose esančius prietaisus (Galvao ir kt., 2011; Kaklauskas, Rute ir kt., 2015). Elektros energijos ištekliai gali būti efektyviai naudojami tuo atveju, jeigu galima nuspėti, kada resursų prireiks ir kada jie liks nepanaudoti arba panaudoti neefektyviai, pavyzdžiui, kai aplinkoje nėra žmonių. Sistemose, veikiančiose pagal nustatytą laiką, kai taikomi prietaisų įjungimo ir išjungimo laikmačiai, negalima įvertinti, ar tikrai reikia panaudoti prietaisą tam tikru momentu. Numatytais laiko intervalais įjungiant prietaisus tiek komerciniuose, tiek gyvenamuosiuose namuose, nepavykta efektyviai panaudoti resursus. Realiomis sąlygomis veikiančios deterministinės sistemos retai įjungia įrenginius tinkamu laiku. Įprastos sistemos neįvertina, ką ir kada naudotojas šioje aplinkoje atliks ir kokiais konkrečiais prietaisais naudosis.

Iš anksto žinomi elektros energijos naudojimo tarifai leidžia sukurti kontrolės įrenginį naudojimo tvarkaraščiui kurti, siekiant užtikrinti perteklinį resursų naudojimą ir išlaidų viršijimą. Tokiu atveju siūloma spręsti optimizavimo uždavinį, kurio išėjimas – optimalus įrenginio arba aplinkos energijos naudojimo planas (Sun ir kt., 2012). Rezultatas gali būti panaudotas tolesnės dienos sąnaudų prognozei atlikti.

Elektros energijos prietaisų naudojimas gali būti valdomas arba nevaldomas išmaniojo būsto aplinkoje. Kontroluojami prietaisai ir jų sąnaudos dažniausiai yra determinuoti ir gali būti valdomi HEM (angl. *Home Energy Management*), t. y. namo energijos valdymo sistemos, priemonėmis.

Viena iš namų energijos valdymo sistemos užduočių – surinkti duomenis iš išmaniojo būsto aplinkoje veikiančių ir elektros energiją galinčių naudoti prietaisų. Tai atliekama galios matuokliais ir jutikliais. Siekiant sumažinti išlaidas ir gauti atsaką iš sistemos tenka pasitelkti poreikio valdymo sistemos priemones, išgaunant poreikio atsaką ir sudarant sąlygas sumažinti elektros energijos sąnaudas bei poreikį. Tokiose sistemose dažniausiai komponuojamos tokios posistemės, kurios leistų taupyti elektros energijos išteklius, atliekant prietaisų valdymą, sąnaudų optimizavimą bei prognozę į ateitį, analizuojant istorinius duomenis (Eriksson, 2012; Hassan ir kt. 2012). Išmaniojo būsto atveju kiekvienas individualus paslaugų sistemos energiją vartojantis prietaisas lemia bendrą suvartojimą. Turint komunikacinę ryšį tarp paslaugų teikėjo ir vartotojo, galima sudaryti abiem naudingą resursų deterministinių sąnaudų valdymo modelį. Nagrinėjamos autonominės energijos išteklių sistemos atveju tai būtų sąlygos vartotojo naudojamoms paslaugoms, t. y. sudaroma galimybė sutaupyti perkeliant panaudojimą į pigesnio tarifo zoną automatiškai, jeigu tai nesumažina vartotojo išmaniojo būsto sistemos teikiamo komforto arba užtikrina, kad komfortas yra pakankamas.

Poreikio atsakas sudaro galimybę, kad nedidelis energijos poreikio sumažinimas gerokai sumažintų elektros energijos kainą. Galimybė sumažinti vartojamos energijos poreikį yra alternatyvi poreikio perstūmimui į vėlesnį arba iš anksto numatant į ankstesnį laiką (angl. *demand shifting*).

Išmaniojo būsto aplinkos paslaugų sistemose yra daug elektros energiją naudojančių įrenginių. Naudojant energijos valdymo sistemą galima sutaupyti valdant energijos sąnaudas, kurios tenka didžiausią dalį energijos suvartojantiems prietaisams,

tačiau būtina suderinti komforto ir taupymo ypatybes.

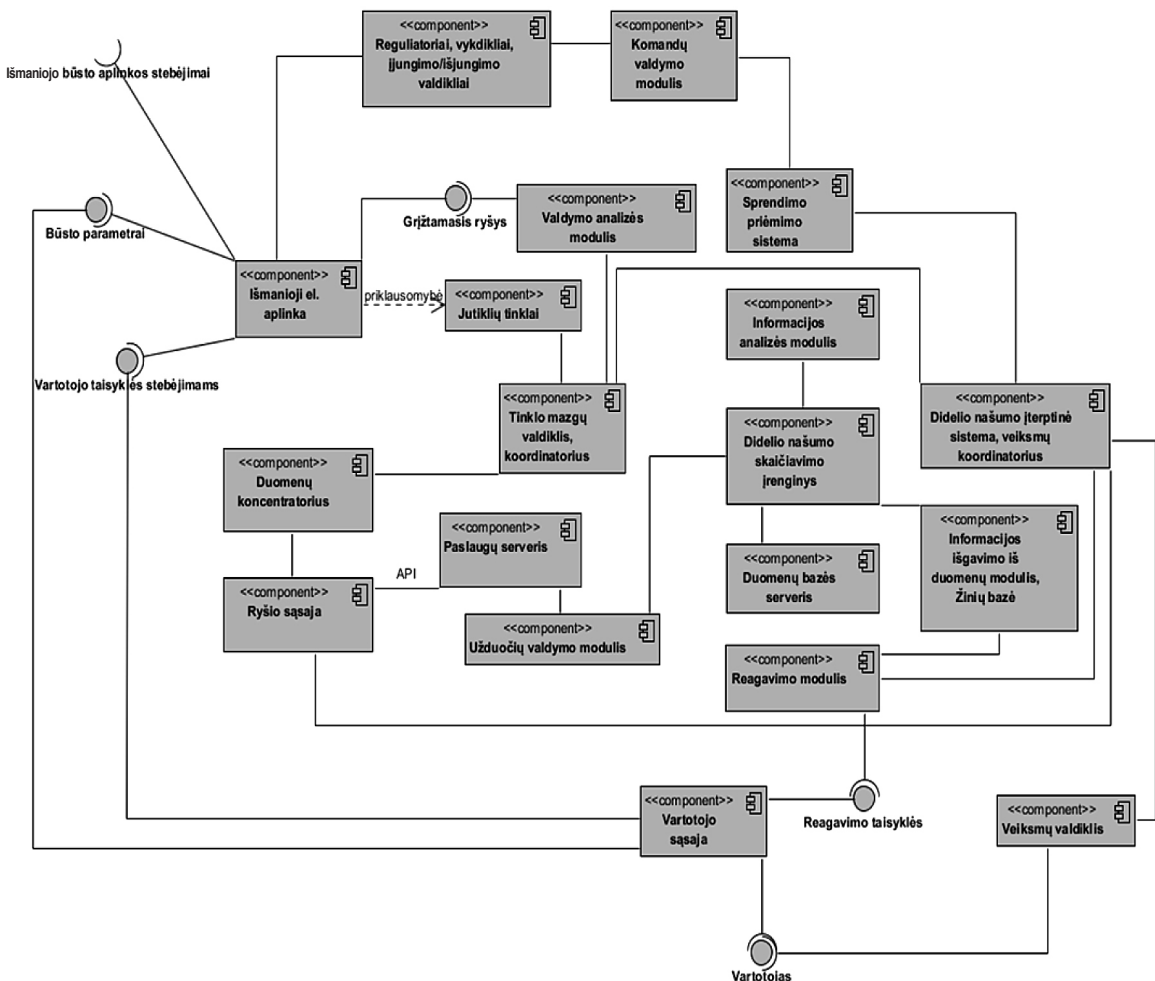
2. Informacinės sistemos išvystymas ir integravimas išmaniojo būsto elektroninių paslaugų sistemoje

Informacinė sistema buvo projektuojama siekiant surinkti energijos suvartojimo duomenis, atlikti jų analizę. Informacinės sistemos funkcijos buvo praplėstos moduliais, kurie leido atlikti elektros sąnaudų prognozę bei užtikrinti autonominią tokios posistemės valdymą (Zulkas ir kt., 2015). Bendra autonominės energijos valdymo posistemės koncepcinė schema pateikiama 1 paveiksle.

Pirminis aplinkos parametrų ir energijos duomenų surinkimas vykdomas jutikliuose. Jutikliai jungiami į sistemą belaidžio tinklo priemonėmis. Jie persiunčia informaciją koncentraciniam įrenginiui (tinklo sąsajos įrenginys), kurio viduje integruota duomenų kaupimo posistemė. Tinklo sąsajos įrenginys atsakingas už duomenų vientisumą ir ryšį su serveriu. Vartotojo aplinkoje surinkti energijos sąnaudų ir kitų matavimų duomenys kaupiami serverio duomenų bazėje.

Elektros energijos sąnaudų stebėsenos posistemės pagrindą sudaro jutikliai, kurie renka duomenis apie energiją suvartojančius įrenginius. Išmaniojo daikto / jutiklio paskirtis – surinkti reikiamus duomenis iš aplinkos ir juos paversti struktūruota informacija (Alipp ir kt., 2006; Jiang ir kt., 2010; Sabit ir kt., 2011). Jo sudėtyje numatyta pirminė talpykla, duomenų filtravimo posistemė, savianalizė veikimo kokybei užtikrinti, taip pat įgalintos ryšio su koncentratoriumi technologijos. Projektuojamoje sistemoje jutikliai naudojami elektros įrenginių sąnaudoms nustatyti.

Elektroninės paslaugos modulio funkcionalumas priklauso nuo teikiamos pa-



1 pav. Duomenų surinkimo ir elektros energijos išteklių valdymo sistemos bendroji architektūra

slaugos infrastruktūros. Paslaugų modulio tikslas – kaupti elektros energijos sąnaudų duomenis ir atlikti prognozę numatant elektros sąnaudų pasiskirstymą tam tikrais laiko intervalais į priekį. Modulį galima sujungti su išmaniojo būsto įranga, siekiant efektyviau panaudoti elektros energijos išteklius. Šio sujungimo sprendimo realizacija aprašyta pateikiant sistemos architektūrinių komponentų schemą (2 pav.). Eksperimentams atlikti elektros energijos išteklių valdymo sistemos prototipe aplin-

kos šviestuvams valdyti realizuotas *Zig-Bee* protokolas ir tokios topologijos tinklas, taip pat *Ethernet TCP/IP* tinklas, skirtas sujungti prietaisus į bendrą komunikacinį tinklą, grindžiant eksperimentą Chen ir kt. (2012) rekomacijomis. Jutiklių tinklams įjungti į komunikacinę sistemą naudojamos technologijos apima pagrindines komunikacines sąsajas, būdingas mikrovaldiklių periferijai valdyti: UART, SPI, I2C ir kt. (2 pav.).

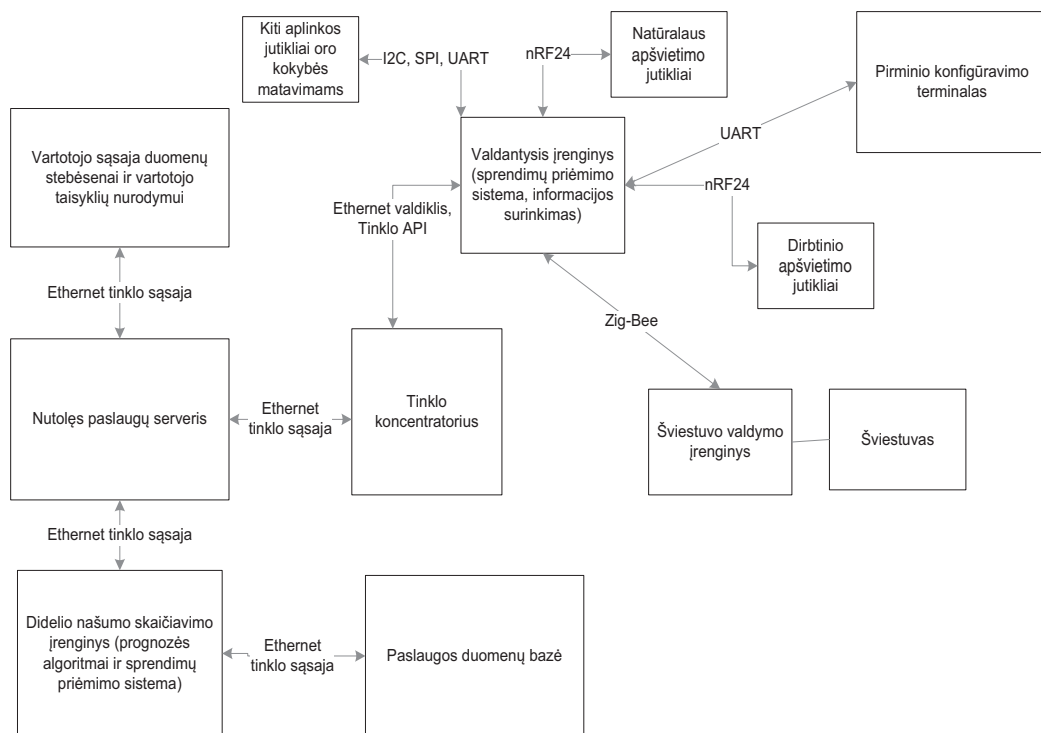
Šie duomenys toliau perduodami prognozės posistemai ir sprendimų priėmimo

sistamai. Nagrinėjama energijos išteklių valdymo sistemos architektūra (2 pav.), kuri, pradėdant aplinkos parametrų nuskaitymo jutikliais būdu, perduoda duomenis sprendimų priėmimo sistemai, veikiančiai neuroninio tinklo generuojamų scenarijų pagrindu, ir leidžia reaguoti į aplinkoje susidariusias situacijas ir priima autonominius valdymo sprendimus.

Sistemoje koncentratorius surenka jautiklių vienmačių arba daugiamatį parametrų duomenis iš aplinkos. Siekiant užtikrinti visavertę sistemos veikimą, būtini algoritmai, kurie leistų įvertinti susidariusias situacijas. Įvertinta susidariusi situacija $s_{sk,t}$ perduodama paslaugos gavėjui. Tokiu būdu vartotojas, gavęs informaciją apie situaciją, siekdamas tam tikro komforto sąlygų ir atsižvelgdamas į individualius poreikius, gali nuotoliniu būdu

reguluoti energiją vartojančius įrenginius. Įvairios paskirties jutiklių duomenys yra vienijami per koncentravimo įrenginius ir siunčiami į paslaugos duomenų bazę. Serveris yra atsakingas už informacijos kaupimą, išgavimą, sudėtingų daugiamatį signalų apdorojimą ir atpažinimą. Kiekvienas jutiklis tinkle komunikuoja belaidžiam tinkle naudodamas telemetrijos technologiją (angl. *machine-to-machine*, arba M2M) ir keičiasi duomenimis su serveriu standartizuotais duomenų perdavimo protokolais. Dėl galimybės komunikuoti su įvairia tarpine ryšio technologine įranga koncentratoriaus įrenginyje būtinas užduočių dispečerizavimas. Viena iš koncentratoriaus funkcijų yra stebėti išmaniuosius daiktus.

Elektroninių paslaugų paketo pagrindinė paskirtis – užtikrinti patogią vartotojo



2 pav. Eksperimentams naudojama energijos išteklių stebėsenos ir valdymo sistemos struktūra

sąsają. Kiekvienai paslaugai kuriamas situacijos stebėjimo ir paslaugos valdymo modulis, kuris gali būti nesiejamas su kitais moduliais, tačiau priklauso tam tikrai modulių grupei. Pagrindinis tokios elektroninių paslaugų teikimo sistemos komponentas yra sprendimų priėmimo sistema.

3. Sprendimų priėmimo sistemoje taikomi sprendimų priėmimo būdai

Sprendimų priėmimo sistema padeda analizuoti realaus laiko ir istorinius duomenis pagal nustatytas paslaugų atlikimo taisykles bei priimti elektrinių prietaisų valdymo sprendimus. Galutinis sprendimas yra įgyvendinamas naudojantis pakoreguotu sprendimo modeliu, kurio taisyklių taikymas vyksta realiu metu keičiant sprendimų priėmimo modelio platformą ir analizuojant sistemos grįžtamojo atsako rezultata (Zulkas ir kt., 2015). Priimant kiekvieną sprendimą siūloma naudotis šiais žingsniais: apibrėžti problemą, apibrėžti reikalavimus, nusistatyti tikslus, apibrėžti kriterijus, pasirinkti sprendimo priėmimo įrankį, pasiūlyti sprendimų alternatyvas, validuoti pasirinktą sprendimą pagal problemos formuluotę.

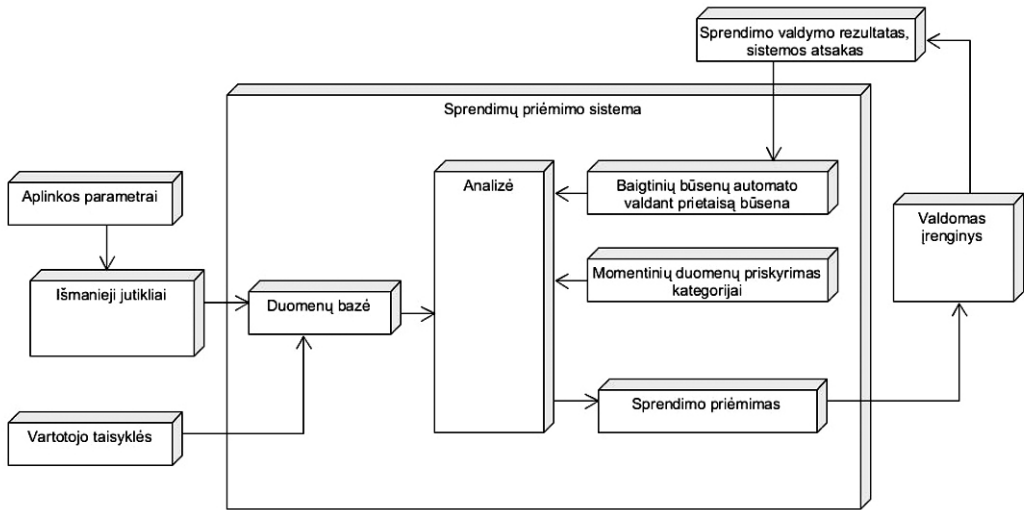
Aprašyta sprendimų priėmimo sistemos struktūra (3 pav.) yra integruojama į paslaugų sistemą įvairaus tipo užduotims spręsti, pavyzdžiui, išmaniojo būsto įvairaus tipo energijos ištekliams taupyti.

Pirmiausia apibrėžiama problema: tai momentas, kai analizuojant aplinkos parametrus netenkinama viena ar kita charakteristika, todėl būtina priimti valdymo sprendimą. Reikalavimų apibrėžimas yra aprašomas pagal esamas valdymo sistemos galimybes (pagal tai, ką ir kaip galima valdyti).

Suformuluojamas tikslas valdyti viešus ar kitus aplinkoje esančius prietaisus, siekiant sutaupyti elektros energijos są-

naudas bei valdyti išmaniojo būsto aplinką. Apibrėžti kriterijai priklauso nuo siektino sprendimo tikslo (išmanios aplinkos parametrų pakeitimo greitis, elektros energijos sąnaudos, ribotos naudojimo galimybės įvertinimas ir kt.). Galimas atvejis, kai priimamas vienas sprendimas, tačiau galimi keli jo variantai. Tokiu atveju pagal kriterijus ir valdymo prioritetus turi būti rekomenduojamas veiksmingiausias (priklausomai nuo istorinių duomenų bei grįžtamojo ryšio rezultato) sprendimas. Sprendimas validuojamas remiantis valdymo rezultatu, kurį sudaro naudojamos galios charakteristikos ir aplinkos parametrų pasikeitimas. Šiame darbe automatiniame sistemos būsenos nustatymui ir koregavimui naudojamas formalizmas, grindžiamas baigtinių būsenų automato aprašymo priemonėmis. Jo būsenos – galimos susidarančios sistemoje situacijos ir iš jų gaunami priimamų sprendimų variantai, kai perėjimai tarp būsenų yra sąlygos, kurios lemia vieno ar kito sprendimo priėmimą. Šioms sąlygoms pakitus, galima priimti kitokią sprendimą. Tokiu būdu aprašant prietaisų reguliavimo modulį, elektrinį įrenginį galima dinamiškai reguliuoti autonominiu būdu.

Kai sprendimo priėmimas yra susijęs su ateities prognozės rezultatais, svarbu prognozuoti energijos suvartojimą ateityje ir šį rezultatą panaudoti reguliavimo sprendimui priimti. Be istorinių duomenų prognozavimo negalima užtikrinti, kad prietaiso sąnaudos neviršys leistinų reikšmių ir sistema galės paskirstyti maksimaliai galimą sąnaudų kiekį per nagrinėjamą intervalą. Siekiant tikslo, kad sistema veiktų autonomiškai, siūloma laikytis tam tikrų apribojimų: (a) išteklių taupymo architektūrai prognozės modulis yra viena iš svarbiausių sistemos komponentų, (b) būtina tiksliai užtikrinti autonominės paslaugos veikimą.



3 pav. *Sprendimų priėmimo sistemos struktūrinė schema*

Sprendimo priėmimo sistemos modelis yra aprašomas pasirinkta programavimo kalba ir jame realizuotos taisyklės nusako vykdymo veiksmų eigą nustatytu konkrečiu reguliavimo atveju. Reguluojamos konfigūracijos sprendimų priėmimo sistemos realizacija pasireiškia tuo, kad veiksmų seka gali būti reguliuojama naudojantis vartotojo aprašytais taisyklėmis ir techniniais bei loginiais ribojimais. Siekiant apibrėžti realaus laiko reguliuojamą konfigūraciją, galima įvertinti šiuos konkretaus jutiklio teikiamus parametrus: minimali reikšmė, maksimali reikšmė, vidutinė reikšmė ir siekiamas rezultatas (-ai).

Baigtinis būsenų automatas yra naudojamas tiksliai sprendimų priėmimo sistemos modelio (taisyklių rinkinio) konfigūracijai aprašyti. Pagal užduotas taisykles, gali būti konfigūruojamas požymių atpažinimas ir sprendimų priėmimo parametrai. Kiekvienas požymis gali būti koreguojamas naudojantis aplinkos parametrų stebėsenos duomenimis, todėl šis sprendimų priėmimo būdas yra save reguliuojantis realiu laiku.

Kartu su prognozės rezultatais sprendimai gali turėti kelis panaudojimo būdus:

- Sprendimai priimami priklausomai nuo prognozės rezultatų (galimas prognozavimo metodas grindžiamas neuroninio tinklo generuojamais prognozavimo scenarijais, pateiktas 4 skyriuje);
- Sprendimai priimami operatyviai įvertinus situaciją, t. y. be prognozės duomenų vertinimo.

Savireguliuojantis baiginių būsenų automatas modelio konfigūracijai yra įgyvendintas mikrovaldiklio įterptinės sistemos branduolyje ir iškeltas (angl. *porting*) į aukštesnio lygio programavimo kalbos valdymo posistemę, kuri pritaikyta autonominiam sistemos veikimui užtikrinti. Norint priimti elektros energijos taupymo sprendimą, galima remtis keliais būdais:

- valdymo komanda atliekama pagal istorinius duomenis,
- valdymo komanda atliekama palyginus su kaimynų (jutiklių, valdyklių) duomenimis,
- atsitiktinio sprendimo siūlymu pagal normalizuotus duomenis.

Valdymo komanda pagal istorinius duomenis pateikia sprendimą ir remiasi vartotojo taisyklėmis, kaip korektiškai valdyti

įrenginių darbą. Palyginimo su kaimynais atveju priimamas sprendimas patikslinamas atsižvelgiant į duomenis iš kitų panašiai veikiančių posistemų. Atsitiktinio sprendimo siūlymas remiasi vidutine valdomo prietaiso sąnaudų reikšme ir gali būti naudojamas be istorinių duomenų, tačiau yra labiausiai priklausomas nuo momentinio valdymo rezultato. Sprendimas yra koreguojamas realiu laiku, atsižvelgiant į gautą rezultatą. Atsitiktinis sprendimas reikalauja, kad sistemos atsakas būtų gautas palyginti greitai, todėl tarpinė grįžtamojo ryšio analizė nebūtų atliekama.

4. Duomenų prognozės scenarijų taikymas MatLab generuojamų neuroninių tinklų priemonėmis

Aprašomas energijos duomenų prognozės būdas, grindžiamas MatLab programinės įrangos kūrėjų siūlomu sprendimu – sukurti neuroninį tinklą NAR, kuris generuotų prognozės scenarijų (MatLab, 2016). Šis būdas leidžia sukurti scenarijus energijos sąnaudų prognozei ir perplanaudoti sąnaudų paskirstymą kuriamoje elektros energijos valdymo sistemoje. MatLab integruotoje aplinkoje, atliekant neuroninio tinklo skaičiavimus, sugeneruojamas kodas arba diagrama, pasitelkiant *MatLab Compiler* įrankiais ir kitomis *MatLab Simulink* kodo generavimo priemonėmis (Al Shamisi ir kt., 2011).

Paslėptų neuronų skaičius tinkle gali būti valdomas ir turi įtaką skaičiavimo efektyvumui bei rezultatų kokybei. Paslėptų neuronų skaičius ir duomenų imties priklausomybė pateikta formule:

$$N_p \leq \frac{N_m \cdot E_t}{N_i \cdot N_{i\bar{s}}};$$

čia N_p – paslėptų neuronų skaičius, N_m – tinklo mokymo imties eilučių skaičius (treni-

ravimo šablonų), E_t – tolerancinė paklaidos riba neuroniniame tinkle. Ši skaičiavimo taisyklė nusako, kada tinklo konfigūracija bus pakankama išiminti nagrinėjamai problemai (MatLab, 2016).

Galima pasirinkti „Simple Script“ arba „Advanced Script“ funkcijas, kad būtų galima sukurti scenarijų, vykdomą per komandinę eilutę arba per grafinę sąsają. Tokiu būdu tinklo darbas tampa valdomas išorinėmis programavimo priemonėmis (MatLab, 2016). Tinklą galima išsaugoti kaip dirbanti *netMatLab* darbo aplinkoje. Galima atlikti papildomus bandymus, pateikiant naujus įėjimo duomenis. MatLab aplinkoje generuotas kodas būna iškart su komentarais, todėl jį nesunku pakoreguoti pagal poreikius.

Prieš vykdant scenarijų įėjimų ir išėjimų duomenų vektoriai turi būti iš anksto apibrėžti ir įkrauti MatLab darbinėje aplinkoje. Jeigu jie nėra apibrėžti, tai juos reikėtų nurodyti rankiniu būdu.

NAR tinklo sukūrimo žingsnyje nurodomi reikiami metodo parametrai. NAR tinklas kuriamas naudojant tiesioginio sklidimo tinklo metodą su pagal nutylėjimą nurodyta tangento sigmoidine (angl. *tansigmoid*) perdavimo funkcija paslėptuose sluoksniuose ir tiesine perdavimo funkcija kituose sluoksniuose (MatLab, 2017). Tinklas turi du įėjimus: išorinį ir grįžtamojo ryšio. Kai tinklas apmokytas, šis ryšys uždaromas. Kiekvienam iš įėjimų sudaroma *tapped* delsimo linija ankstesnėms reikšmėms saugoti. Kad NAR tinklui būtų priskirta tinkama architektūra, reikia nurodyti delsimo linijas kiekvienam įėjimui ir nurodyti paslėptų sluoksnių skaičių:

```
inputDelays = 1:4;
feedbackDelays = 1:4;
hiddenLayerSize = 10;
```



```
net = narnet(inputDelays,feedbackDelays,hiddenLayerSize);
```

čia *inputDelays* – įėjimų delsimo linija, *feedbackDelays* – grįžtamojo ryšio delsimo linija, *hiddenLayerSize* – paslėptų neuronų skaičius, *net* – neuroninis tinklas.

Didinant neuronų skaičių arba delsimo linijas, skaičiavimo trukmė ilgėja, tačiau didėja tikimybė adekvačiau permokyti tinklą. Esant didesniai neuronų skaičiui galima išspręsti sudėtingesnes užduotis. Didesnis paslėptų neuronų sluoksnių skaičius taip pat leidžia spręsti sudėtingesnes netiesines problemas. Kad būtų naudojamas daugiau nei vienas paslėptas neuronų sluoksnis, reikia nurodyti paslėptų neuronų sluoksnių dydžius kaip masyvo elementus *fitnet* komandoje.

Jeigu tinklas apmokomas naudojant delsimo linijas, tai visos jos turi turėti pradines reikšmes (naudojamos įėjimų ir išėjimų pradinės reikšmės). Reikšmėms užpildyti naudojama komanda *preparets*. Ši funkcija turi tris įėjimo parametrus: *tinkle*, įėjimo ir išėjimo vektorius. Funkcija gražina pradines reikšmes, reikiamas užpildyti delsimo linijų reikšmėms:

```
[inputs,inputStates,layerStates,targetSeries]= preparets(  
net,inputSeries,{},targetSeries);
```

čia *net* – neuroninis tinklas, *inputSeries* ir *targetSeries* – įėjimo ir išėjimo vektoriai.

Duomenų paskirstymas tarp tinklo kūrimo procesų atliekamas naudojant *trainRatio*, *valRatio*, *testRatio* nustatymus, įėjimo ir išėjimo vektoriai atsitiktinai paskirstomi procentiškai pagal nurodytas reikšmes kiekvienam procesui: 70 % mokymams, 15 % validavimui, 15 % testams (galimų reikšmių intervalas 0-1).

Tinklo mokymas naudojant algoritmą, nustatomą pagal nutylėjimą: *Levenberg-*

Marquardt algorithm (trainlm). Didesnių užduočių sprendimui galima naudoti Bajeso reguliarizacijos algoritmą su komanda *trainbr* arba Gradientinių skalių (angl. *Scaled Conjugate Gradient*) algoritmą su komanda *trainscg*:

```
net.trainFcn = 'trainbr'; (Bajeso reguliacija).
```

```
net.trainFcn = 'trainscg'; (Gradientinis skalių metodas).
```

Tinklo mokymas pradamas po komandos:

```
[net,tr] = train(net,inputs,targets,inputStates,layerStates);
```

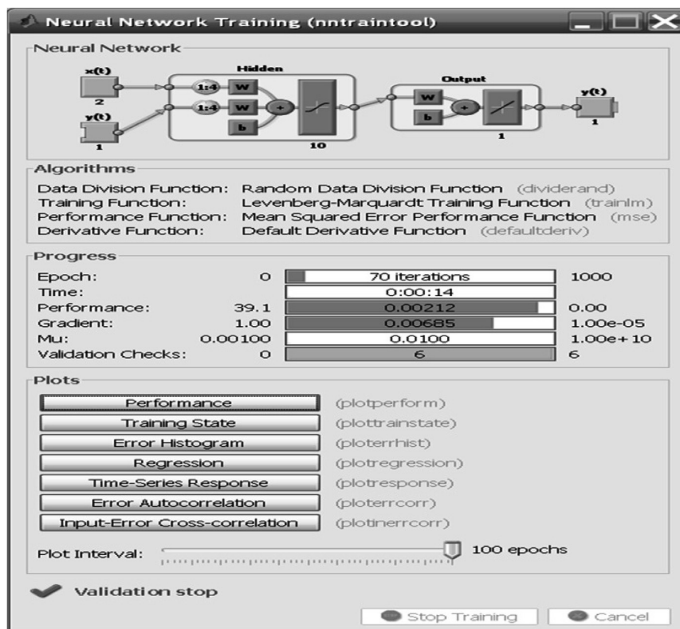
Kaip atrodo mokymo langas su realiame laike besikeičiančiais rezultatais bei galimybe peržiūrėti sujungimo sąsajas, parodo 4 paveikslas.

Kai tinklas yra apmokytas, jį galima naudoti išėjimams skaičiuoti. Pateiktame kode skaičiuojami tinklo išėjimai, klaidos ir bendras efektyvumas.

```
outputs = net(inputs,inputStates,layerStates); % išėjimas  
errors = gsubtract(targets,outputs);  
% klaidos  
performance = perform(net,targets,outputs) % bendras efektyvumas
```

Visas mokymas yra atliekamas naudojantis atviru ciklu, dar vadinamu nuoseklialygiagrečia architektūra (angl. *series-parallel architecture*). Standartinė eiga yra vykdoma siekiant sukurti tinklą atviru ciklu ir tik po apmokymo (įskaitant validavimą ir testavimą) sukurti ciklo išeities sąlygą, kad prognozę būtų galima vykdyti daugeliu žingsnių į ateitį (angl. *multistep-ahead prediction*).

Pateiktame kode uždaromas ciklas NAR tinkle. Jeigu ciklas neuždaromas, galima prognozuoti tik vieną žingsnį į priekį.



4 pav. Neuroninio tinklo mokymo langas

Šaltinis: neuroninis tinklas sudarytas aplinkoje (MatLab, 2016).

Prognozuojama sekanti reikšmė sekai $y(t)$ iš ankstesnių $y(t)$ reikšmių. Jeigu ciklas uždaromas, prognozė gali būti panaudota prognozuoti daugiau žingsnių, nes naujos $y(t)$ reikšmės bus naudojamos vietoje atieties reikšmių.

```
netc = closeloop(net);
netc.name = [net.name ' - Closed
Loop'];
view(netc)
[xc,xic,aic,tc] = preparets(netc,inputsSeries,{},targetSeries);
yc = netc(xc,xic,aic);
perfc = perform(netc,tc,yc)
```

Ištrinamas delsimas, kad prognozės rezultatas būtų gautas vienu žingsniu anksčiau.

```
nets = removedelay(net);
nets.name = [net.name ' - Predict One
Step Ahead'];
view(nets)
```

```
[xs,xis,ais,ts] = preparets(nets,
inputSeries,{},targetSeries);
ys = nets(xs,xis,ais);
earlyPredictPerformance =
perform(nets,ts,ys)
```

Neuroninis tinklas turės mažesnes delsimo linijas po delsimo ištrynimo. Tinklo išėjimas tokiu atveju bus $y(t+1)$ vietoje $y(t)$.

Jeigu tinklo efektyvumas netenkina, galima bandyti šiuos žingsnius:

- Iš naujo apibrėžti pradinis tinklo svorius ir poslinkius naudojantis komanda *init*.
- Padidinti paslėptų neuronų sluoksnių skaičių.
- Padidinti mokymosi vektorių skaičių.
- Padidinti įėjimo reikšmių skaičių, jeigu prieinama daugiau informacijos.
- Panaudoti kitą mokymosi algoritimą.

Neuroniniai tinklai grindžiami netiesiniais algoritmais ir dirbtinio intelekto

galimybėmis. Priklausomai nuo pasirinktų neuroninio tinklo ir duomenų dažnio charakteristikų generuojamas atitinkamas prognozės rezultatas gali būti skirtingas, todėl gauti griežtą paklaidos įvertį yra sunku. Kiekvieną kartą apmokytas tinklas turi naujus svorius bei poslinkius. Neuroniniais tinklais įvertinti energijos sąnaudas ir jas nuspėti, kokios jos bus ateityje, leidžia sistemos architektūroje numatomas istorinių duomenų saugojimas ir informacijos išgavimo iš duomenų modulis, kuriam būtina didelio našumo aparatinė skaičiavimo įranga. Priešingu atveju ribojami apdorojamų duomenų kiekiai. Siekiant naudoti neuroninius tinklus su nagrinėjamu duomenų masyvu, reikia rinktis mažesnę prognozės reikšmių intervalą, o kad jį būtų galima išlaikyti, reikia mažinti duomenų dažnį, pritaikant skaitmeninius duomenų apdorojimo filtrus.

Išvados

Darbe išvystyta informacinės sistemos infrastruktūra taikoma išmaniojo būsto paslaugos realizacijai, kai reikia užtikrinti duomenų surinkimą iš belaidžiam tinkle veikiančių įrenginių. Aprašyta elektros energijos išteklių valdymo sistemos archi-

tektūra išskirtinė tuo, kad suteikia automatinio elektros prietaisų reguliavimo galimybę: įrenginių valdymo sprendimai priimami panaudojant istorinius galios sąnaudų duomenis ir aplinkos parametrų momentines reikšmes. Realaus laiko duomenų rinkimo tinklo duomenys gali padėti nuspėti ir reguliuoti elektros energijos ateities sąnaudas. Energijos valdymo sistema gali būti pritaikyta naudoti ne tik energijos suvartojimo duomenis, bet ir papildomai įvertinti žmogaus buvimą ir elgseną aplinkoje, aplinkos temperatūrą arba kitus paslaugai priklausančius išmaniuosius jutiklius.

Sprendimų priėmimo sistemą rankiniu būdu sukonfigūruoti reikia tik kartą, pateikiant vartotojo taisykles ir paslaugos bei naudojimo ribojimus kuriamam ir apmokomam neuroniniam tinklui. Pasiūlyta sistemos architektūra, skirta išmaniosios aplinkos parametrams kaupti ir stebėti bei įrenginių reguliavimo sprendimams realiu laiku priimti. Atsižvelgiant į energijos taupymo sistemos panaudojimą išmaniojo būsto valdymo sistemos architektūroje, siekiama ištirti esamas ir pasiūlyti naujas galimybes autonomiškai valdyti elektros energijos sąnaudas.

LITERATŪRA

ADINYA, O. J.; DAOLIANG, L. (2012). Transceiver energy consumption models for the design of Low power wireless sensor networks. *Research and Development (SCoReD)*, p. 193–197.

AL SHAMISI, M. H.; ASSI, A. H.; HEJASE, H. A. N. (2011). Using MATLAB to Develop Artificial Neural Network Models for Predicting Global Solar Radiation in Al Ain City – UAE. *Engineering Education and Research Using MATLAB*.

ALIPP, C.; GALPERTI, C. (2006). An Adaptive Maximum Power Point Tracker for Maximising Solar Cell Efficiency in Wireless Sensor Nodes. *Circuits and Systems*, p. 3722–3725.

ANDZIULIS, A.; BIELSKIS, A. A.; DENISOVAS, V.; RAMAŠAUSKAS, O.; BIELSKIENĖ, J.;

BIELSKIS, P.; GUSEINOVĖ, E. (2009). An Approach of Creating of an Intelligent Mobile Tutoring Eco-social Laboratory for Assisted Recreation. *Electronics and Electrical Engineering* (ISSN 1392–1215), Nr. 3(91) p. 61–66.

BARNAWI, Y. A.; KESHTA, M. I. (2016). Energy Management in Wireless Sensor Networks Based on Naive Bayes, MLP, and SVM Classifications: A Comparative Study. *Journal of Sensors*, vol. 2016, p. 1–12. Prieiga per internetą: <<http://dx.doi.org/10.1155/2016/6250319>>.

BIELSKIS, A. A.; DZEMYDIENĖ, D.; DENISOV, D.; ANDZIULIS, A.; DRUNGILAS, D. (2009). An approach of multi-agent control of bio-robots using intelligent recognition diagnosis of persons

with moving disabilities. *Technological and Economic Development of Economy*, vol. 15, no. 3, p. 377–394.

CHEN, S.; YAO, J.; WU, Y. (2012). Analysis of the Power Consumption for Wireless Sensor Network Node Based on Zigbee. *Procedia Engineering*, 29, p. 1994–1998.

CHEN, Y.; CHUAH, C. N.; ZHAO, Q. (2008). Network configuration for optimal utilization efficiency of wireless sensor networks. *Ad Hoc Networks*, 6, p. 92–107.

DZEMYDIENĖ, D.; GUSEINOVIEŅĒ, E.; ŽULKAS, E. (2015). An approach of designing of decision support system in smart human closed environment. In *Proceedings of International Conference Social Innovations for Global Growth*, p. 26–27.

ERIKSSON, N. *Predicting demand in district heating systems – A neural network approach*. Uppsala University, 2012.

GALVAO, J.; LEITAO, S.; MALHEIRO, S.; GAIO, T. (2011). Model of Decentralized Energy on Improving the Efficiency in Building Services. *Energetics (IYCE)*, p. 1 – 8.

HASSAN, M.; BERMAK, A. (2012). Solar Harvested Energy Prediction Algorithm for Wireless Sensors. *Quality Electronic Design (ASQED)*, p. 178–181.

JIANG, N.; CHEN, Z. (2010). Model-Driven Data Cleaning for Signal Processing System in Sensor Networks. *Signal Processing Systems (ICSPS)*, p. 237–242.

KAKLAUSKAS, A.; ZAVADSKAS, E. K.; CERKAUSKAS, J.; UBARTE, I.; BANAITIS, A.; KRUTINIS, M.; NAIMAVICIENE, J. (2015). Housing Health and Safety. Decision Support System with Augmented Reality. *Procedia Engineering*, 122, p. 143–150.

KAKLAUSKAS, A.; RUTE, J.; ZAVADSKAS, E. K.; DANIUNAS A.; PRUSKUS V.; BIVAINIS J.; GUDAUSKAS R.; PLAKYS, V. (2015). Passive House Model for Quantitative and Qualitative Analyses and Its Intelligent System. *Intelligent Systems Reference Library*, 81, p. 87–112. December 2015. DOI: 10.1007/978-3-319-13659-2_3.

MatLab, Introduction: Frequency Domain Methods for Controller Design, 2015 [interaktyvus], [žiūrėta 2016-03-10]. Prieiga per internetą: <<http://ctms.engin.umich.edu/CTMS/index.php?example=Introduction§ion=ControlFrequency>>.

MatLab, Nonlinear autoregressive neural network with external input. 2017 [interaktyvus], [žiūrėta 2017-01-16], Prieiga per internetą: Nonlinear autoregressive neural network with external input.

MatLab, Neural Network Time-Series Prediction and Modeling [interaktyvus], [žiūrėta 2016-04-07]. Prieiga per internetą: <<https://se.mathworks.com/help/nnet/gs/neural-network-time-series-prediction-and-modeling.html?requestedDomain=www.mathworks.com>>.

SABIT, H.; AL-ANBUKY, A.; HOSSEINI, H. G. (2011). Wireless Sensor Network Based Wildfire Hazard Prediction System Modeling. *Procedia Computer Science* 5, p. 106–114.

Sun, H. C.; Huang, Y. C. (2012). Optimization of Power Shedding for Energy Management in Smart Homes. *Procedia Engineering*, 38, p. 1822–1827.

ZULKAS, E.; ARTEMCIUKAS, E.; DZEMYDIENE, D.; GUSEINOVIEŅĒ, E. (2015). Energy consumption prediction methods for embedded systems. *Ecological Vehicles and Renewable Energies (EVER)*, p. 1–5. Prieiga per internetą: <<http://dx.doi.org/10.1109/EVER.2015.7112932>>.

THE DEVELOPMENT OF AN INFORMATION SYSTEM FOR SUSTAINING ENERGY CONSUMPTION IN SMART HOMES

Dalė Dzemydienė, Evaldas Žulkas

S u m m a r y

Information systems play an important role in smart home management. They collect monitoring data for devices connected to a wireless network (e.g., sensors, lights, thermometers). Such systems are classified as complex modern electronic services management systems that can provide environmental management services in a smart home. However, in order for smart devices to be able to use energy efficiently, additional software modules are needed to

make these devices more efficiently used. The purpose of our research work is to develop the functions of the information system, enabling the estimation of electrical energy consumption in the smart housing system and the use of forecasting data for controlling electricity costs.

Keywords: information system, Internet of Things, smart house management services, electric power management system.

2017 m. gegužės 29 d.