



KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS

Darius Ruiba

**KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS
SIURBLIO SISTEMOS VALDYMO METODO SUKŪRIMAS IR
TYRIMAS**

Baigiamasis magistro projektas

Vadovas

Doc. dr. Renaldas Urniežius

KAUNAS, 2015

KAUNO TECHNOLOGIJOS UNIVERSITETAS
ELEKTROS IR ELEKTRONIKOS FAKULTETAS
AUTOMATIKOS KATEDRA

**KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS
SIURBLIO SISTEMOS VALDYMO METODO SUKŪRIMAS IR
TYRIMAS**

Baigiamasis magistro projektas
Valdymo technologijos (kodas 621H66001)

Vadovas

Doc. dr. Renaldas Urniežius

Recenzentas

Projektą atliko

Darius Ruiba

KAUNAS, 2015



KAUNO TECHNOLOGIJOS UNIVERSITETAS

Elektros ir elektronikos

(Fakultetas)

Darius Ruiba

(Studento vardas, pavardė)

Valdymo technologijos 621H66001

(Studijų programos pavadinimas, kodas)

Baigiamojo projekto „Pavadinimas“
AKADEMINIO SAŽININGUMO DEKLARACIJA

20 ____ m. ____ d.
Kaunas

Patvirtinu, kad mano **Dariaus Ruibos** baigiamasis projektas tema „Kombinuotos saulės kolektorių ir šilumos siurblio sistemos valdymo metodo sukūrimas 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)

Ruiba, D. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos valdymo metodo sukūrimas ir tyrimas. *Magistro* baigiamasis projektas / vadovas doc. dr. Renaldas Urniežius; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas, Automatikos katedra.

Kaunas, 2015. 67 psl.

SANTRAUKA

Magistrantūros baigiamąjį darbą sudaro 67 puslapiai. Darbe yra 5 lentelės, 34 paveikslai, 2 priedai. Darbui panaudoti 22 literatūros šaltiniai.

Pagrindinis darbo tikslas – sukurti kombinuotą saulės kolektoriaus ir šilumos siurblio sistemos valdymo modelį atsižvelgiant į šilumos siurblio ir saulės kolektorių kontūro parametrus.

Šiam tikslui pasiekti buvo išskirti uždaviniai:

1. Išnagrinėti techninėje literatūroje pateiktus tipinius saulės kolektorių ir šilumos siurblių veikimo principus, pagrindines charakteristikas ir matematinius modelius.
2. Išstudijuoti skirtumus tarp skirtingų šilumos siurblio valdymo galimybių: kai galia yra nereguliuojama ir kai galia yra reguliuojama.
3. Sukurti ir patikrinti programos algoritmą, kuris parinks šilumos siurblio kompresoriaus parametrus bei optimaliausią valdymo režimą.

Darbe atlikta esamų saulės kolektorių ir šilumos siurblių tipų, jų veikimo principų ir konstrukcijų analizė. Pateikti statistikos duomenis apie saulės šiluminės energijos potencialą Lietuvoje.

Magistriniame darbe aprašomi metodai, kurie padeda parinkinėjant optimizavimo kriterijus, sprendžiant optimalaus valdymo uždavinius ir problemas. Nurodomi saulės kolektoriaus, šilumos siurblio dalių: garintuvo ir kondensatoriaus, bei akumuliacinės talpos šilumos balanso lygčių sudarymo principai ir išvestos balanso lygtys. Pateikiama pagrindinio algoritmo schema, pagal kurią buvo ieškomas optimalus šilumos siurblio kompresoriaus galingumas, jo darbo režimai skirtingomis aplinkos sąlygomis. Patikrintas algoritmo teisingumas įvedant žinomų sistemų parametrus.

Šio tyrimo rezultatus galima pritaikyti optimalus valdymo uždavinių sprendime, kai reikia parinkinėti kombinuotos sistemos elementus ir suderinti jų darbą.

Reikšminiai žodžiai: kombinuotos sistemos valdymas, saulės kolektorius, šilumos siurblys, šilumnešis

Ruiba, D. Development and Investigation of the Control System Method for Combined Heat Pump and Evacuated-Tube Solar Collectors System. Final *Master Thesis Project* / supervisor doc. dr. Renaldas Urniežius; Kaunas University of Technology, Faculty of Electrical and Electronics Engineering, Department of Automation

Kaunas, 2015. 67 pages

SUMMARY

First of all, the volume of this Master Thesis is 67 pages. It is composed of 5 tables and 34 images and 2 attachments. This thesis was written based on 22 sources of reference.

The main objective of this thesis was to develop and investigate the Control System Method for Combined Heat Pump and Evacuated-Tube Solar Collectors System.

In order to achieve the above mentioned objective the following tasks were created:

1. To analyse the typical operating principles of solar collectors and heat pumps, as well as their main characteristics and mathematical models.
2. To study the differences among different heat pump operating options: when power is controlled and uncontrolled.
3. To create and test program algorithm which would select best parameters and the most suitable optimal control mode for the heat pump compressor.

The thesis investigated the existing types of solar collectors and combined heat pumps, their operating principles and constructions. Also, statistical data regarding the potential of solar thermal energy use in Lithuania was provided.

This Master thesis includes descriptions of methods which are applicable during the process of choosing optimization criteria, as well as, for solving various tasks and problems related to optimal control. Also provided are derived equations and principles of writing energy balance equations for evacuated-tube solar collector, boiler and heat pump parts, such as evaporator and condenser. Additionally provided is the main algorithm scheme, which was used to determine the optimal heat pump compressor capacity and its operating modes in different environmental conditions. Finally, the effectiveness and accuracy of the created algorithm was tested by entering settings that are used for existing systems.

Summarizing, it can be stated that the results of this thesis research are applicable for solving optimal control tasks when elements for combined system need to be selected and their working principles need to be mutually adjusted.

Key words: combined system control, solar collector, heat pump, heat transfer medium

TURINYS

IVADAS	8
1. LITERATŪROS APŽVALGA	9
1.1. Saulės spindulinė energija ir apšvieta	9
1.2. Saulės kolektorių tipai	12
1.2.1 Plokščiasis saulės kolektorius	12
1.2.2 Vakuuminis saulės kolektorius.....	15
1.2.3 Tūrinis saulės kolektorius	17
1.3. Šiluminiai procesai vykstantys saulės kolektoriuje	18
1.3.1 Saulės kolektoriaus pagrindinės lygtys	19
1.4. Šilumos siurbliai	21
1.4.1 Šilumos siurblys ir jo veikimo principas.....	21
1.4.2 Šilumos siurblių tipai	25
1.5. Pontriagino minimumo (maksimumo) principas	26
2. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS MODELIO SUDARYMAS	29
2.1. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos struktūrinė schema	29
2.2. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos šiluminis balansas 29	
2.3. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos optimizavimo kriterijaus pasirinkimas	33
2.4. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos „Matlab Simulink“ modelio sudarymas	35
3. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS ALGORITMO IR PROGRAMOS SUKŪRIMAS	36
3.1. Kompresoriaus galia pastovi, režimas „Įjungta/Išjungta“, laiko intervalo kitimas kas dt	36
4. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS MODELIO ELEMENTŲ PRADINIŲ PARAMETRŲ PASIRINKIMAS IR APSKAIČIAVIMAS	40
4.1. Saulės kolektoriaus parinkimas	40
4.2. Saulės kolektoriaus šilumnešio parinkimas	41
4.3. Saulės kolektoriaus ir jo šilumos kontūro, kuriame teka polipropilenglikolio ir vandens mišinys, visuminės šiluminės talpos apskaičiavimas	41

5. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS MODELIO EKSPERIMENTINIS TYRIMAS.....	42
5.1. Optimalios galios kompresoriaus suradimas pasirinktai kombinuotajai sistemai.....	42
5.2. Kompresorių darbo režimų palyginimas, kai kompresorių galia yra skirtinga.....	44
5.3. Šilumnešio atitekančio į saulės kolektorių vidutinės temperatūros kaita per 10 h, esant skirtingoms E_n vertėms	46
5.4. Kompresoriaus darbo režimų palyginimas, kai sistema modeliuojama 10 h, o kompresorius gali būti įjungiamas arba išjungiamas kas 0,5 h, taip pat keičiant T_v reikšmę	47
5.5. Kompresoriaus parinkimas, kai kinta saulės apšvieta E_s ir lauko temperatūra T_{lauko}.....	49
5.6. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos palyginimas su sistema, kurioje yra tik saulės kolektorius.....	52
IŠVADOS IR PASIŪLYMAI	56
LITERATŪRA	57
PRIEDAI	59
1 PRIEDAS. WOLFRAM MATHEMATICA 10“ PROGRAMOS ALGORITMAS MATEMATINIŲ IŠRAIŠKŲ RADIMUI	60
2 PRIEDAS. PROGRAMOS KODAS PARAŠYTAS C# KALBA.....	62

ĮVADAS

Atsinaujinantys energijos šaltiniai tampa vis svarbesni šiuolaikinio žmogaus gyvenime. Vis daugiau valstybių, įmonių ir organizacijų prisideda prie subalansuotos ateities energetikos vystymosi. Orientuojamasi į vietinių ir atsinaujinančių energijos šaltinių įsisavinimą. Tuo pačiu visuomenė tampa sąmojingesnė ir pati siekia sumažinti šiltnamio efektą skatinančių dujų išsiskyrimą, išsaugoti švarią aplinką, kurioje gyvena, todėl daugiau dėmesio ir lėšų skiriama atsinaujinančių energijos šaltinių vystymui ir vartojimui.

Lietuvoje beveik nėra įprastinių energijos šaltinių: dujų, naftos, akmens anglies, tad didžioji suvartojamos energijos dalis yra įvežtiniai energijos šaltiniai. Iškastinio kuro kainos nuolat didėja, nes mažėja jo resursai, todėl atsinaujinantys energijos šaltiniai tampa vis svarbesni ir perspektyvūs.

Saulės energija yra galingiausia ir pati perspektyviausia atsinaujinančios energijos rūšis, kuri pasiekia žemę milijardus metų. Saulės spinduliavimo galia pasiekianti žemę yra $1,9 \cdot 10^{17}$ W, o saulės metinis energijos kiekis yra 3900 mln. PJ per metus, tačiau tik nedidelė dalis spindulinės energijos gali būti panaudota [1]. To priežastis - žemės paviršių pasiekusi saulės spindulinė energija yra išsklaidyta ir nepastovi.

Lietuvoje mažosios saulės kolektorių sistemos karšto vandens ruošimui plačiausiai naudojamos nedideliuose gyvenamuose namuose ir sodybose [2]. Lietuvoje saulė vidutiniškai šviečia apie 1800 h per metus, o vidutinis suminis energijos potencialas yra 1000 kWh/m² [2]. Šie rodikliai artimi Vokietijos, Danijos ir kitų Europos valstybių rodikliams, kur plačiai naudojamos mažosios saulės kolektorių sistemos vandeniui pašildyti. Gerai įrengtos saulės kolektorių sistemos gali padengti apytiksliai nuo 15 % iki 60 % metinių šiluminės energijos sąnaudų, skirtų karšto vandens ruošimui ir patalpų šildymui [2].

Tačiau, kaip dar padidinti saulės šiluminės energijos pasisavinimą naudojant saulės kolektorius? Mokslinė straipsniai [3] teigia, kad padidinti saulės kolektoriaus efektyvumą įmanoma, jei kolektorius bus naudojamas kartu su šilumos siurbliu.

Lietuvoje ir pasaulyje atliekami kombinuotų saulės kolektorių ir šilumos siurblio sistemų bandymai. Tačiau susiduriame su problema, kad dažnai siurbliai parenkami neatsižvelgiant į saulės kolektoriaus ir visos sistemos parametrus arba tas pasirinkimas būna netinkamas ir neargumentuotas. Todėl ši tema aktuali ir verta nagrinėjimo.

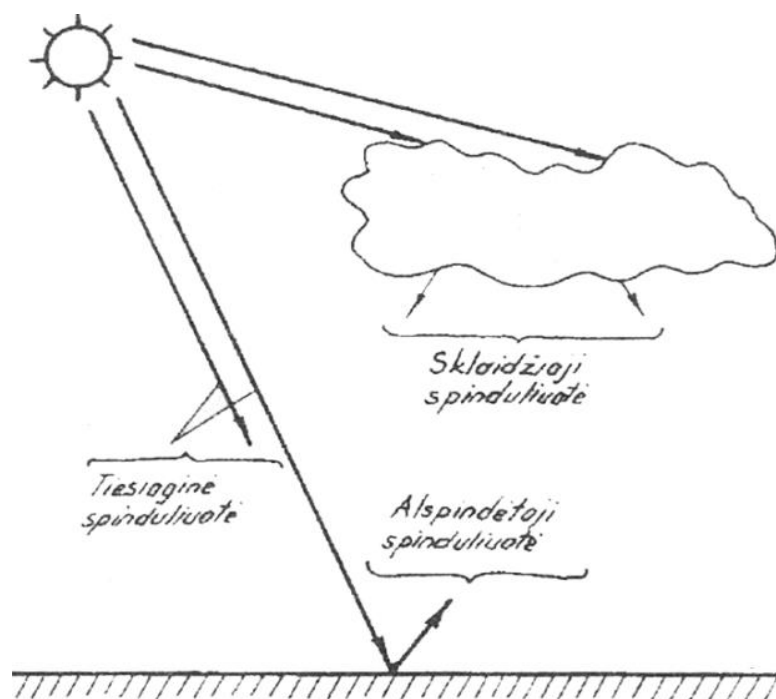
Šiame magistriniame darbe bus kuriamas ir tiriamas kombinuotos saulės kolektorių ir šilumos siurblio sistemos valdymo metodas. Kad būtų įgyvendintas šis tikslas bus išanalizuota techninė ir mokslinė literatūra, išnagrinėti saulės kolektoriaus ir šilumos siurblio veikimo principai. Planuojama darbe pateikti optimalaus sistemos parametro pasirinkimo metodiką, programos algoritmą, kuris suras optimalią šilumos siurblio galią ir jo valdymo režimą, esant skirtingiems sistemos parametrams.

1. LITERATŪROS APŽVALGA

1.1. Saulės spindulinė energija ir apšvieta

Saulė yra pats galingiausias atsinaujinantis energijos šaltinis. Švari, neišsenkanti saulės energija pasiekia žemę milijardus metų ir pasieks ateityje. Teorinis metinis saulės energijos potencialas yra tūkstančius kartų didesnis už kitų rūšių energijos potencialą [2].

Saulės spindulinę energiją charakterizuoja saulės spinduliuotė. Saulės spinduliuotė sudaro: tiesioginę, sklaidžią ir atspindėtoji spinduliuotės (1.1 pav.).



1.1 pav. Tiesioginė, sklaidžioji ir atspindėtoji saulės spinduliuotė [2]

Tiesioginė spinduliuotė patenka nuo saulės disko. Šios spinduliuotės spindulių kritimo į žemės paviršių kampas yra žinomas, tačiau dalis tiesioginių saulės spindulių yra išsklaidomi dėl debesų ir atmosferos dulkių poveikio. Šie spinduliai krenta į žemės paviršių įvairiais kampais ir vadinami sklaidžiąja spinduliuote [2]. Dalis saulės spindulių, kurie pasiekia žemę ar kitus paviršius atsispindimi, o tokia spinduliuotė vadinama atspindėtąja [2]. Tiesioginės ir sklaidžiosios spinduliuočių suma vadinama pilnutine spinduliuote [2].

Saulės spinduliuotės energetines savybes nusako saulės energinė apšvieta - G ir saulės spindulinė energija - S [2].

Saulės energinė apšvieta G nurodo spinduliuotės galią, tenkančią vienam kvadratiniam metrui paviršiaus ploto - W/m^2 [2].

Saulės spindulinė energija S – tai saulės spindulinės energijos kiekis, tenkantis vienam kvadratiniam metrui paviršiaus per tam tikrą laikotarpį. Šios energijos dimensija yra J/m^2 arba kWh/m^2 .

Lietuvoje vienam kvadratiniam metrui žemės paviršiaus tenka apie 3,6 GJ energijos per metus. Tai nėra didelis kiekis palyginus su Afrika (10 GJ), Graikija (5,4 GJ) ar kitomis valstybėmis, esančiomis arčiau pusiaujo, tačiau metinis saulės energijos kiekis Lietuvoje yra toks kaip Lenkijoje, Latvijoje ar Vokietijoje [2]. Pagrindinė priežastis, dėl ko saulės energija Lietuvoje 1 m žemės paviršiaus ploto siekia tik 3,6 GJ yra ta, kad saulės energija patenkanti į žemės paviršių išsisklaido žymiai didesniame paviršiaus plote negu tose geografinėse platumose, kur vidurdienį saulė stovi zenite [2]. Saulės spinduliai taip pat nukeliauja ilgesnį atstumą atmosferoje, kur patiria žymiai didesnius absorbcijos ir difuzijos nuostolius [2]. Lietuvoje vidutinis metinis saulės energijos kiekis, kuris pasiekia horizontalų 1 m^2 paviršiaus plotą, yra apie 1000 kWh/m^2 [2].

1.1 lentelė. Vidutinis saulės energijos kiekis tenkantis 1 m^2 horizontalaus paviršiaus plotui Lietuvoje kiekvieną mėnesį, skirtingoje vietoje, kWh/m^2 [2]

Vietovė	Mėnesiai												Saulės energija per metus
	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	
Nida	14	31	72	108	155	171	165	148	97	54	17	10	1042
Šilutė	15	33	72	106	154	169	161	143	96	55	16	9	1029
Lazdijai	19	37	74	103	150	159	153	142	94	56	20	13	1021
Telšiai	14	32	69	104	154	168	156	141	96	55	19	11	1018
Kybartai	20	37	74	103	148	157	152	140	93	56	21	14	1015
Klaipėda	12	31	67	102	155	168	161	147	94	53	16	8	1013
Šiauliai	13	31	68	100	154	163	153	142	94	53	17	9	996
Dotnuva	10	29	70	103	154	164	158	145	92	50	11	3	989
Vėžaičiai	13	32	67	104	153	154	155	140	94	53	14	8	988
Kaunas	16	33	70	99	146	155	150	138	90	52	16	9	976
Utena	15	33	67	96	145	151	147	133	84	50	16	8	946
Vilnius	16	34	69	93	142	146	142	136	84	50	17	10	939
Varėna	11	31	68	94	147	152	147	142	85	48	11	3	939
Dūkštas	9	28	65	97	150	157	153	137	84	47	10	1	938
Biržai	8	27	65	96	148	156	151	135	83	46	9	2	926

Pagal 1 lentelės duomenis Lietuvai tenka vidutinis saulės energijos kiekis į 1 m horizontalaus paviršiaus plotą, kuris Europoje siekia 830 - 1100 kWh/m² [2]. Esant giedram orui Lietuvoje, pilnutinė horizontalaus paviršiaus apšvieta vasaros vidurdienį būna 810 - 850 W/m² [2]. Rudenį saulės apšvietos vidurkis yra apie 400 W/m² [2].

1.2 lentelėje pateikta horizontalaus paviršiaus apšvieta saulės spinduliais Kauno mieste vidurdienį saulės laiku, kW/m² [2].

1.2 lentelė. Horizontalaus paviršiaus apšvieta saulės spinduliais Kauno mieste vidurdienį saulės laiku, kW/m² [2]

Apšvietos tipas	Mėnesiai											
	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII
Tiesioginė apšvieta, esant giedram dangui	0,15	0,28	0,44	0,63	0,68	0,72	0,77	0,62	0,46	0,33	0,18	0,10
Visuminė apšvieta, esant giedram dangui	0,23	0,39	0,56	0,75	0,81	0,85	0,81	0,75	0,57	0,42	0,25	0,16
Tiesioginė apšvieta, esant vidutiniam debesuotumui	0,03	0,08	0,17	0,19	0,28	0,32	0,27	0,25	0,18	0,09	0,02	0,02
Visuminė apšvieta, esant vidutiniam debesuotumui	0,12	0,22	0,37	0,42	0,55	0,59	0,54	0,48	0,37	0,21	0,09	0,08

Dar viena saulės spinduliuotės energetinė charakteristika yra saulės spindėjimo trukmė. Tai laikotarpis, kai saulės apšvieta viršija 0,21 kW/m². Saulės spindėjimo trukmė Lietuvos teritorijoje yra skirtinga - Lietuvos vakarinėje dalyje siekia net 1900 h per metus, o rytinėje dalyje - apie 1650-1700 h per metus [2]. Saulės spindėjimo trukmė Kaune pateikta 1.3 lentelėje.

1.3 lentelė. Saulės spindėjimo trukmė Kaune [2]

Dydis	Mėnesiai												Metinis kiekis
	I	II	III	IV	V	VI	VII	VIII	IX	X	XI	XII	
Spindėjimo Trukmė, h	41	59	133	178	255	177	267	234	166	96	37	30	1773
Dienų „be saulės“ skaičius	18	13	8	4	2	1	1	2	3	9	19	21	101

Iš pateiktų lentelių matyti, kad, Lietuvoje galima naudoti saulės energiją šilumai išgauti, tačiau yra pakankamai neigiamą įtaką darančių veiksnių, kurie silpnina šios energijos panaudojimą. Vienas iš pagrindinių ir neišvengiamų veiksnių yra gamtos reiškiniai, tokie kaip nakties laikas, rudens ir žiemos periodas, kurių metu saulės energijos Lietuvoje galima panaudoti

mažiausiai [2]. Tokiais mėnesiais, kaip gegužė, birželis ar liepa, saulės energijos potencialas Lietuvoje būna didžiausias ir tai galima išnaudoti tinkamai, paverčiant saulės energiją reikiama energija buityje ar pramonėje [2].

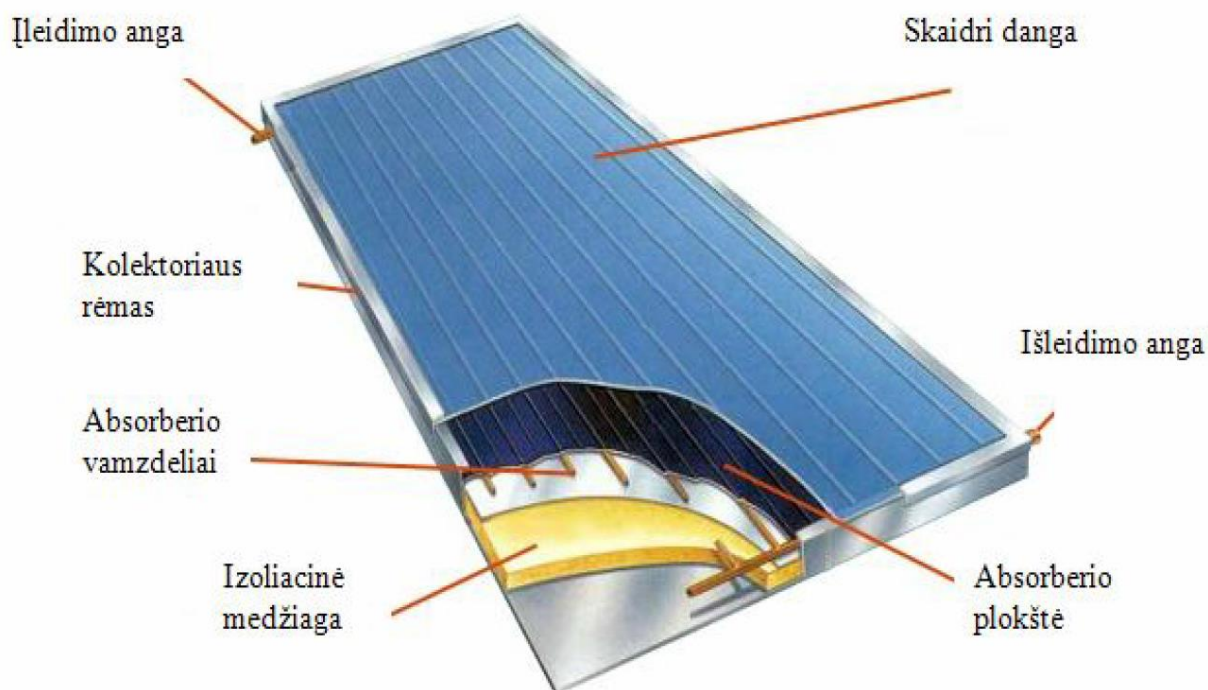
Bendras saulės energijos kiekis Lietuvoje yra pakankamai didelis, kadangi Lietuva pakliūna į taip vadinamą anomalijos zoną. Šiam reiškiniiui ypatingai palankus yra šiaurinė ir vakarinė Lietuvos dalis, kadangi čia metinė saulės spinduliavimo energija siekia net 1300 kWh per metus, todėl Lietuva yra pakankamai perspektyvi valstybė pagal saulės energiją, kurioje galima plėtoti šiluminės energijos gamybą iš saulės kolektorių [2].

1.2. Saulės kolektorių tipai

1.2.1 Plokščiasis saulės kolektorius

Lietuvoje plačiausiai paplitęs plokščiasis saulės kolektorius. Šio kolektoriaus paskirtis - saulės energijos pavertimas šilumine energija. Plokščiasis kolektorius yra palyginti paprastos konstrukcijos. Standartinis šio tipo kolektoriaus vieno modulio efektyvus plotas A (matavimo vienetai - m^2) yra $0,8 m^2 - 2,5 m^2$ [2].

Plokščiojo kolektoriaus pagrindiniai elementai pateikti 1.2 paveiksle.



1.2 pav. Plokščiojo saulės kolektoriaus vandeniui šildyti schema [4]

Plokščiojo saulės kolektoriaus pagrindiniai elementai [5]:

- Absorberis, kuris sugeria saulės spinduliuotės energiją ir perduoda ją cirkuliuojančiam šilumnešiui, kuris paduodamas į kolektorių per įleidimo angą ir išleidžiamas iš kolektoriaus per išleidimo angą.
- Skaidri dangą, kuri pralaidi saulės spinduliams ir tuo pačiu sumažinanti konvekcinius ir šiluminius nuostolius.
- Izoliacinė medžiaga, skirta sumažinti kolektoriaus šiluminius nuostolius.
- Kolektoriaus rėmas (korpusas), skirtas absorberiu, skaidriai dangai ir izoliacijai sumontuoti, bei kolektoriaus tvirtinimui prie statybinių konstrukcijų.

Svarbiausias saulės kolektoriaus elementas yra absorberis. Absorberį sudaro absorberio plokštė ir absorberio vamzdeliai. Šie plokščiojo saulės kolektoriaus elementai yra gaminami iš medžiagų, kurios yra laidžios šilumai, atsparios aukštai temperatūrai ir korozijai. Dažniausiai naudojamas absorberio medžiagos - vario, aliuminio, nerūdijančio plieno lydiniai bei polimerinės medžiagos. Viena iš plačiausiai naudojamų medžiagų yra varis, todėl, kad jo laidumas šilumai yra apie 2 kartus didesnis už aliuminio ir apie 8 kartus didesnis už plieno [5]. Vamzdelių ekvivalentinis skersmuo paprastai būna 10 - 15 mm [2]. Variniam absorberiu naudojamą lakšto storis būna apie 0,3 - 0,5 mm, o atstumas tarp vamzdelių 100 - 150 mm storio [2]. Aliuminio absorberiu atstumas tarp vamzdelių būna 90 - 140 mm, o 1,0 - 1,2 mm storio plieno absorberiu 60 - 90 mm. Absorberiuose iš polimerinių medžiagų atstumai tarp vamzdelių būna nedideli [2].

Jei absorberis varinis, tai jo šiluminio kontūro šilumnešis gali būti vanduo, tačiau vanduo turi būti filtruojamas. Visgi, dėl platesnio temperatūrų diapazono dažniausiai, kaip šilumnešis yra pasirenkami organiniai neužšalantys skysčiai – pavyzdžiui – polipropilenglikolis. Vandens ir polipropilenglikolio mišinys gali būti naudojamas nerūdijančio plieno arba plastikiniuose absorberiuose. Korozijai absorberiuje sumažinti naudojami specialūs inhibitoriai [6].

Absorberiai gaminami štapavimo, valcavimo, presavimo, virinimo ir litavimo būdu [5]. Kai vamzdeliai prilituoti ar privirinti, jie liečiasi prie lakšto mažu paviršiumi, todėl tokios konstrukcijos absorberio šiluminė kokybė nedidelė. Labai svarbu, kad vamzdelis prie lakšto betarpiškai liestųsi dideliu paviršiumi, tada geriau nuvedama šiluma nuo įkaitusio absorberio paviršiaus [5].

Viena iš svarbiausių saulės kolektorių absorberio charakteristikų yra paviršiaus selektyvumas. Absorberio dangos selektyvumu vadinamas jos absorbavimo gebos trumpabangiam diapazone santykis su emisijos koeficientu ilgabangiam diapazone [2].

Tyrimais nustatyta, kad 96 % saulės spindulių sukonzentruota bangų ilgyje iki 2,5 μm ir 99 % atspindėtų spindulių nuo kolektoriaus absorberio paviršiaus yra didesniame nei 2,5 μm bangų

ilgyje [5]. Tai yra pagrindinė priežastis, kodėl absorberio paviršius padengiamas juoda arba selektyvia danga, kuri geriausiai sugeria ir mažiausiai atspindi krentančius į kolektoriaus paviršių saulės spindulius.

Selektyvumas išreiškiamas santykiu [2]:

$$\alpha / \varepsilon \quad (1)$$

Čia:

α - absorbcijos koeficientas, diapazone 0,25 – 2,5 μm ;

ε - emisijos koeficientas, diapazone 3 – 30 μm .

Selektyvioms dangoms, kai temperatūra 150 ÷ 160°C, selektyvumas santykis yra $\alpha / \varepsilon \approx 8 \div 10$, o juodiems dažams - $\alpha / \varepsilon \approx 1$ [2].

Plačiausiai naudojamas „juodasis“ chromas. Paviršiai, kurių padengimui yra naudojamos šios medžiagos, pasižymi labai geru absorbcijos koeficientu (0,96) ir žemu emisijos koeficientu (0,017 - 0,12), taip pat jie yra ilgaamžiai, atsparūs drėgmei ir aukštai temperatūrai [5].

Maksimali temperatūra, kurią gali pasiekti kolektoriaus absorberis, kai jame esantis šilumnešis necirkuliuoja yra 70°C - atviriems absorberiams, 100°C - uždariems neselektyviems absorberiams ir 190°C - uždariems selektyviems absorberiams, todėl atitinkamai parenkama kolektoriaus skaidri danga bei šiluminė izoliacija [6].

Paprastai saulės kolektoriaus skaidri danga įrengiama 20 – 25 mm atstumu nuo absorberio. Naudojamos skaidrios dangos medžiagos: polietilenas, polichlorvinilas, polietilentereftalatas (PET) ir organinis stiklas. Kolektoriams, kurie eksploatuojami žiemos laikotarpiu, naudojamas dvigubas grūdintas stiklas, numatant tarpus linijiniam stiklo plėtimuisi [2].

Plokščiojo saulės kolektoriaus korpusas gali būti pagamintas iš aliuminio, plieno, įvairių polimerinių mišinių. Poliuretanai, putų polistirolas, akmens, stiklo arba mineralinė vata yra naudojami saulės kolektoriaus korpuso šiluminei izoliacijai, kurios storis 5 – 10 cm.

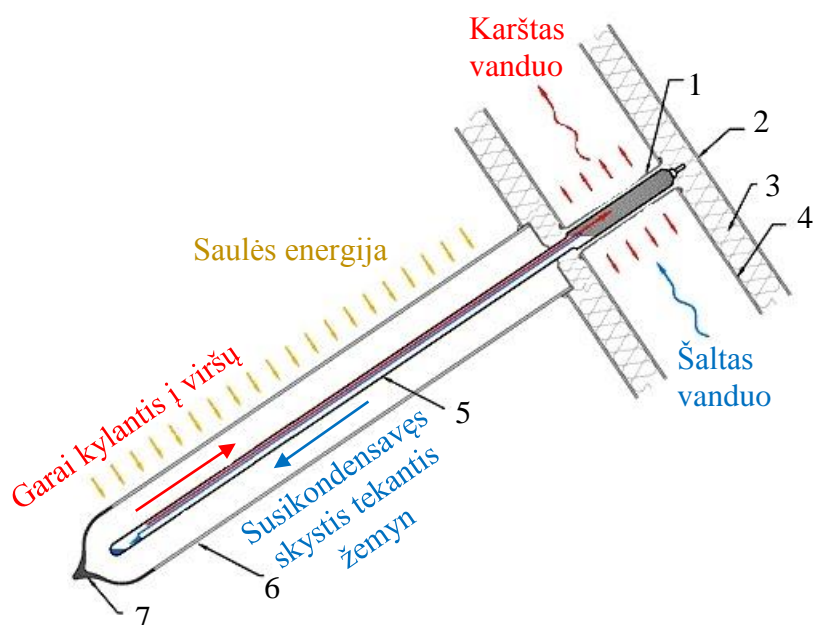
Plokštieji saulės kolektoriai, skirti vandeniui pašildyti, gaminami taip, kad būtų nesunku juos įdiegti tiek ant žemės tiek ant pastato stogo.

Plokštieji saulės kolektoriai montuojami į mažo, vidutinio ir didelio dydžio saulės kolektorių sistemas. Tokios sistemos papildomos šiluminę energiją akumuliuojančiomis talpomis. Naudojamos priverstinės ir natūralios cirkuliacijos sistemos [7]. Plokštieji saulės kolektoriai plačiausiai naudojami Lietuvoje bei kaimyninėse šalyse. Tai lėmė palyginus neaukšta jų kaina ir pakankamai geri energetiniai rodikliai.

1.2.2 Vakuuminis saulės kolektorius

Lietuvoje vis dažniau pradedami naudoti vakuuminiai saulės kolektoriai. Šie saulės kolektoriai gaminami nuo 0,5 m² iki 3 m² ploto moduliais ir jungiami į didesnes saulės kolektorių sistemas.

Vakuuminio saulės kolektoriaus konstrukcijų būna įvairių rūšių: pilnojo vakuumo, dalinio vakuumo, „Heat Pipe“, „U - Pipe“ [5]. 1.3 paveiksle pateikta „Heat Pipe“ vakuuminio saulės kolektoriaus schema [8], o 1.4 paveiksle pateikta „U - Pipe“ vakuuminio saulės kolektoriaus schema [9].



1.3 pav. „Heat Pipe“ vakuuminio saulės kolektoriaus schema [8]

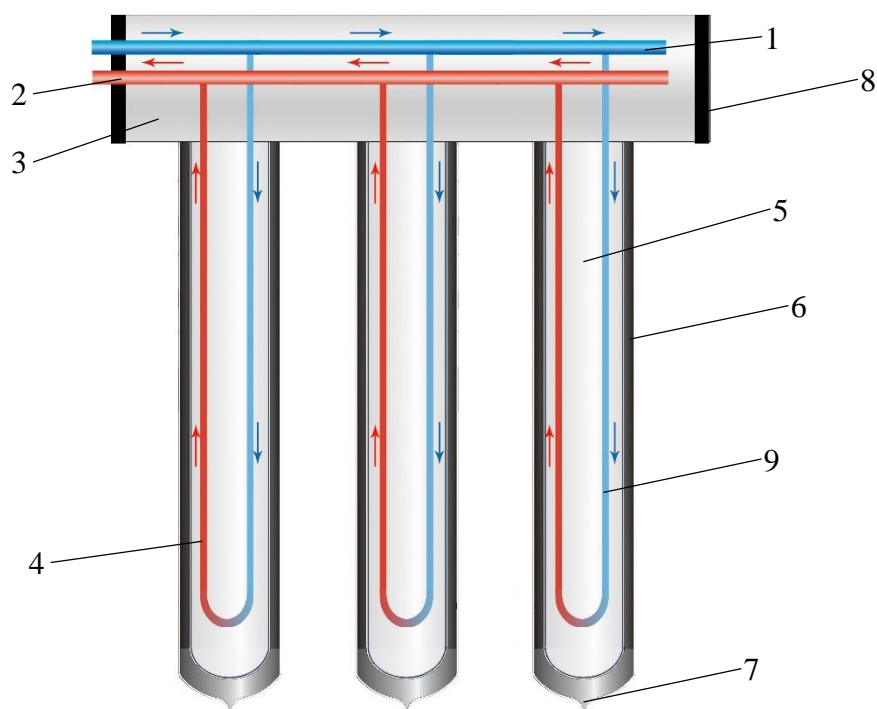
Čia:

1 - kondensėris, 2 – korpusas, 3 – šiluminė izoliacija, 4 – pagrindinis šilumnešio vamzdis, 5 – šilumos vamzdelis („Heat Pipe“), 6 – vakuuminė stiklo kolba, 7 – vakuumo indikatorius.

Pagrindiniai vakuuminių saulės kolektorių elementai yra du specialaus stiklo vamzdeliai: skaidrus išorinis apie 2 cm skersmens ir vidinis apie 1 cm skersmens. Ertmė tarp abiejų vamzdelių yra vakuumuota nuo 0,2 iki 1 Pa slėgio [5]. Vidinis vamzdelis padengtas selektyvi danga, kuri ir yra absorberis. Šio vamzdelio viduje teka šilumnešis. Šio kolektoriaus absorberis turi reguliuojamą polinkio kampą, kuris nustatomas sumontavus kolektorių ant statybinės konstrukcijos [5].

Vakuuminių saulės kolektorių absorberijoje cirkuliuojančio šilumnešio temperatūra pakyla iki 150 – 200 °C ir daugiau [5]. Kolektoriai į vientisą sistemą montuojami iš atskirų modulių, kurie parenkami pagal projektuojamos sistemos parametrus.

Vakuuminiuose saulės kolektoriuose nėra įprastinio korpuso ir šiluminės izoliacijos, kuri yra naudojama tūriniuose ir plokščiuosiuose saulės kolektoriuose. Kadangi sistemoje yra vakuumas, todėl pasiekama geriausia šiluminė izoliacija, t.y. konvekciniai bei šilumos laidumo nuostoliai tokiuose kolektoriuose yra žymiai mažesni nei plokščiuosiuose ar tūriniuose saulės kolektoriuose. Dėl šio konstrukcinės ypatybės vakuuminių kolektorių naudingumo koeficientas pakankamai aukštas ir yra apie 0,84 [5]. Vakuomo užtikrinimas lemia sudėtingesnį šių kolektorių gamybos procesą, todėl šie kolektoriai yra brangesni už plokščiuosius ir tūrinius saulės kolektorius.



1.4 pav. „U - Pipe“ vakuuminio saulės kolektoriaus schema [9]

Čia:

1 – įtekančio į kolektorių šilumnešio kontūras, 2 – ištekančio iš kolektoriaus šilumnešio kontūras, 3 – šiluminė izoliacija, 4 – varinis U formos vamzdelis, 5 – aliuminio plokštelė, 6 – vakuuminė stiklo kolba, 7 – vakuumo indikatorius, 8 – korpusas, 9 – šilumnešis.

Vakuuminiuose kolektoriuose gali būti sumontuoti cilindriniai fokusuojantys reflektoriai – aliumininės ar kitokio metalo plokštelės. Šios konstrukcijos saulės kolektorių reflektoriai fokusuoja saulės spindulinę energiją ir nukreipia ją į vakuuminio absorberio apatinę bei šonines dalis, todėl šildomas visas absorberio paviršius [5]. Tokios konstrukcijos vakuuminiai saulės kolektoriai yra mažiau jautrūs saulės spindulių kritimo kampo kitimui, todėl jie gali sugerti didesnę

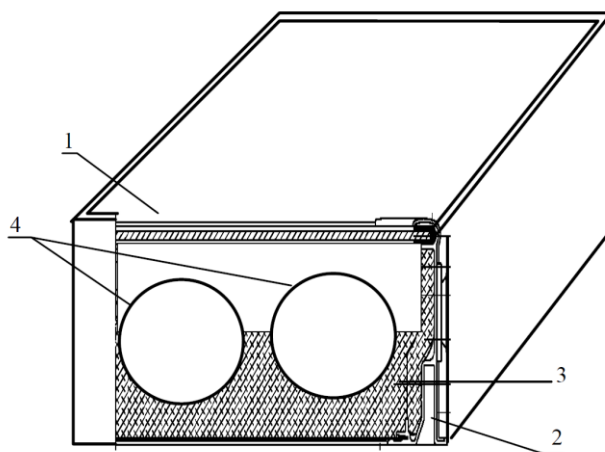
saulės spindulinės energijos kiekį per dieną nei naudojant plokščiuosius arba tūrinius saulės kolektorius.

Tačiau vakuuminiai kolektorai su fokusuojančiais reflektoriais skirti naudoti vietovėse, kur žemės paviršių pasiekia didesnis kiekis tiesioginės saulės spinduliuotės. Lietuvoje didžioji dalis žemės paviršių pasiekusios saulės spinduliuotės yra išsklaidyta, todėl tokios konstrukcijos saulės kolektorai efektyvumas sumažėja [5].

1.2.3 Tūrinis saulės kolektorius

Tūriniai saulės kolektorai yra paprastos konstrukcijos, todėl yra sąlyginai pigūs. Jie gali sušildyti nedidelį vandens kiekį. Šiltas vanduo iš tokio kolektoriaus turi būti sunaudotas tą pačią dieną [1,10].

Tūrinių dar kitaip vadinamų „Breadbox“ tipo saulės kolektorių absorberis sutapatintas su akumuliaciniu vandens rezervuaru [5]. Vandens rezervuaras, kurio paviršius absorbuoja saulės spinduliuotę, patalpintas korpuse su šilumą izoliuojančiomis sienelėmis. Korpusas turi vieno arba dviejų sluoksnių skaidrią dangą [10]. Šių saulės kolektorių viduje įrengiamas vienas arba keli rezervuarai vandeniui. Vandens rezervuarai padengiami juoda selektyvia danga arba selektyviais juodais dažais [10]. Šie kolektorai gaminami pramoniniu būdu, tačiau populiarūs ir savos gamybos. Tūrinio saulės kolektoriaus schema pateikta 1.5 paveiksle.



1.5 pav. Tūrinio saulės kolektoriaus konstrukcija [5]

Čia:

1 - skaidri danga, 2 - korpusas, 3 - šiluminė izoliacija, 4 - vandens rezervuarai.

Tūrinių (kompaktinių) saulės kolektorių rezervuaro tūris yra 40 - 50 l/m² skaidrios dangos plotui. Tokie kolektorai tinka nedideliam iki 100 - 150 litrų vandens kiekiui pašildyti [11]. Tūriniai saulės kolektorai montuojami ant žemės ir skirti individualiam naudojimui. Dažnai šie

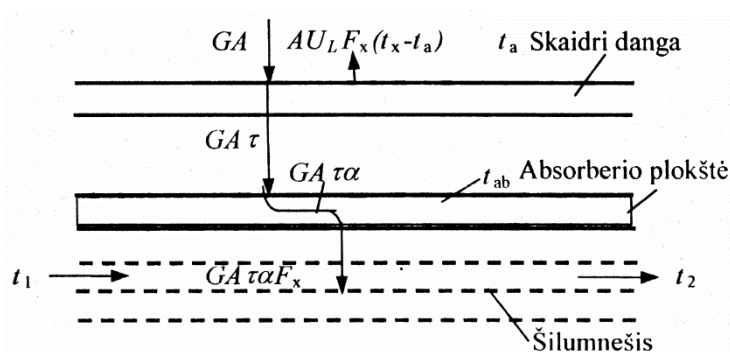
saulės kolektoriai įrengiami vasarnamiuose bei kaimo sodybose, kur nėra didesnio šilto vandens poreikio buitiniams reikmėms [11]. Vandens temperatūra šiuose kolektoriuose pakyla iki 60°C ir, jei vanduo nesunaudojamas, per naktį atvėsta iki 40°C esant aplinkos temperatūrai 20°C [10]. Tūriniai saulės kolektoriai skirti vandeniui pašildyti tik šiltuoju metų laikų. Žiemos metu tokie kolektoriai neeksploatuojami, kadangi juose negalima naudoti įvairių neužšalantių skysčių [10].

1.3. Šiluminiai procesai vykstantys saulės kolektoriuje

Plokščiasis saulės kolektorius saulės spindulinę energiją konvertuoja į šiluminę energiją. Tam, kad kolektorius gerai veiktų, labai svarbu pastatyti jį tinkamu kampu į horizontalią plokštumą, kad į jo skaidrią dangą kristų maksimalus saulės spindulinės energijos kiekis. Šis polinkio kampas vasarą turi būti 15 - 45°, žiemą 50 - 90°, o visiems metams 30 - 60°. Saulės kolektorius turi būti orientuotas į pietus. Nuokrypis nuo pietų krypties $\pm 20^\circ$ didesnės reikšmės neturi, tačiau esant didesniems azimuto kampams, krentančios į kolektoriaus paviršių saulės energijos kiekis žymiai sumažėja. Didesnė Lietuvos dalis yra tarp 54 - 56 paralelės, todėl saulės kolektoriaus pasvirimo kampas turi būti 45 - 55° [1,2].

Saulės spinduliai per skaidrią dangą patenka ant juodos absorberio plokštės ir ją įšildo.

Šiluminė energija saulės kolektoriuje nuvedama nuo absorberio paviršiaus šilumnešiu, tekančiu absorberio kanalais. Skaidri danga sulaiko įšilusio absorberio skleidžiamus ilgabangius spindulius bei įšilusį orą kolektoriuje. Šilumos srautai saulės kolektoriuje pateikti 1.6 paveiksle [2].



1.6 pav. Šilumos srautai saulės kolektoriuje [2]

Esant apšvitai G , į kolektoriaus skaidrią dangą, kurios plotas A , krenta saulės spinduliuotė, kurios galia GA . Esant skaidrios dangos optinio laidumo koeficientui τ , į kolektoriaus vidų per skaidrią dangą pateks saulės spinduliuotės dalis, kurios galia $GA\tau$. Absorberio plokštė, turinti absorbcijos gebą α , sugeria saulės spinduliuotės dalį, kurios galia

$GA\tau\alpha$, ir šios plokštės temperatūra pasiekia t_{ab} reikšmę. Įkaitusi absorberio plokštė šilumnešiui atiduoda šiluminę galią $GA\tau\alpha F_x$ ir absorberio plokštės vamzdeliais tekančio vandens temperatūra pakyla nuo t_1 iki t_2 reikšmės [1,2].

Vidutinė vandens temperatūra absorberio plokštėje randama pagal tokią formulę [1,2]:

$$t_v = 0,5 (t_1 + t_2) \quad (2)$$

Įkaitusi absorberio plokštė dalį šilumos laidumo, konvekcijos ir spinduliavimo būdu atiduoda aplinkai, kurios temperatūra t_a . Jei U_L – suminis šilumos nuostolių koeficientas, perskaičiuotas į skaidrios dangos plotą A ($W/(m^2 \cdot ^\circ C)$), tai šilumos nuostoliai saulės kolektoriuje bus $AU_L F_x (t_x - t_a)$. Dydis F_x priklauso nuo to, kokią temperatūrą pasirenkame t_x (kolektoriaus absorberio temperatūra).

Tuo atveju, kai $t_x = t_{ab}$, tai $F_x = 1$. Ši išraiška praktikoje nenaudojama, todėl, kad nežinoma absorberio plokštės temperatūra.

Kai $t_x = t_v$ (absorberio plokštės temperatūra lygi vidutinei vandens temperatūrai), tai $F_x = F_m$. Šis dydis vadinamas absorberio efektyvumo faktoriumi.

Kai $t_x = t_1$ (absorberio plokštės temperatūra lygi įtekančio į kolektorių vandens temperatūrai), tai $F_x = F_r$. Šis dydis vadinamas absorberio šilumos atidavimo faktoriumi.

Dydis F_m priklauso nuo skysčio debito ir kanalo konstrukcijos ir paprastai būna 0,87 - 0,95 ribose. Absorberio efektyvumo faktorius F_m paprastai būna 1,05 - 1,15 karto didesnis už absorberio šilumos atidavimo faktoriaus reikšmę F_r [1,2].

1.3.1 Saulės kolektoriaus pagrindinės lygtys

Kolektoriaus gaunama galia apskaičiuojama pagal formulę [1,2]:

$$P_1 = GA \quad (3)$$

Jei saulės spinduliai krenta statmenai skaidrios dangos paviršiui, tai kolektoriaus optinio efektyvumo koeficientas [1,2]:

$$\tau\alpha = (\tau\alpha)_n \quad (4)$$

Kolektoriaus šilumnešiui atiduodama galia randama pagal formulę [1,2]:

$$P_2 = GA(\tau\alpha)_n F_m - AU_L F_m (t_v - t_a) \quad (5)$$

Kolektoriaus efektyvumo koeficientas randamas iš formulės [1,2]:

$$\eta = F_m (\tau\alpha)_n - F_m U_L ((t_v - t_a)/G) \quad (6)$$

Ši lygtis gauta laikant, kad šilumos nuostoliai yra proporcingi temperatūrų skirtumui $(t_v - t_a)$. Tačiau dažnai ši priklausomybė nėra tiesinė. Tuomet [1,2]:

$$F_m U_L = a + b (t_v - t_a) \quad (7)$$

Tada:

$$\eta = F_m (\tau\alpha)_n - a ((t_v - t_a)/G) - b ((t_v - t_a)^2 / G) \quad (8)$$

Lygtyse 7 ir 8 naudojami konkretūs koeficientai. Šios lygtys gali būti užrašomos sekančiai [1,2]:

$$\eta = f((t_v - t_a)/G) \quad (9)$$

Ši kolektoriaus energetinė charakteristika yra pagrindinė, ir yra pagrindinis sertifikavimo parametras [1,2].

Sandauga $(\tau\alpha)_n$ yra vadinama kolektoriaus optinio efektyvumo koeficientu.

Kolektoriaus charakteristika $\eta = f((t_v - t_a)/G)$ nustatoma eksperimentiškai, kai tiesioginiai saulės spinduliai krenta statmenai skaidriai dangai.

Realiai per dieną spindulių kritimo kampas kinta. Tuo atveju įvedamas koeficientas K_θ , iš kurio dauginamas dydis $(\tau\alpha)_n$. Koeficiento K_θ priklausomybė nuo spindulių kritimo kampo nustatomos iš lentelių [1,2].

Tada naudingumo koeficientas gali būti užrašomas [1,2]:

$$\eta = F_m (\tau\alpha)_n K_\theta - F_m U_L ((t_v - t_a)/G) \quad (10)$$

Skaičiuojant paprastai pasirenkama, kad sklaidžiosios ir atspindėtosios spinduliuotės kritimo kampas į horizontalų paviršių yra 60° . Esant sklaidžiajai spinduliuotei ir esant viengubai skaidriai dangai, dydis $(\tau\alpha)$ yra apie 1,17 karto mažesnis negu esant statmenai tiesioginei spinduliuotei [1,2].

Minimali saulės apšvieta, kuriai esant, kolektorius pradeda šildyti šilumnešį, užrašoma sekančiai [1,2]:

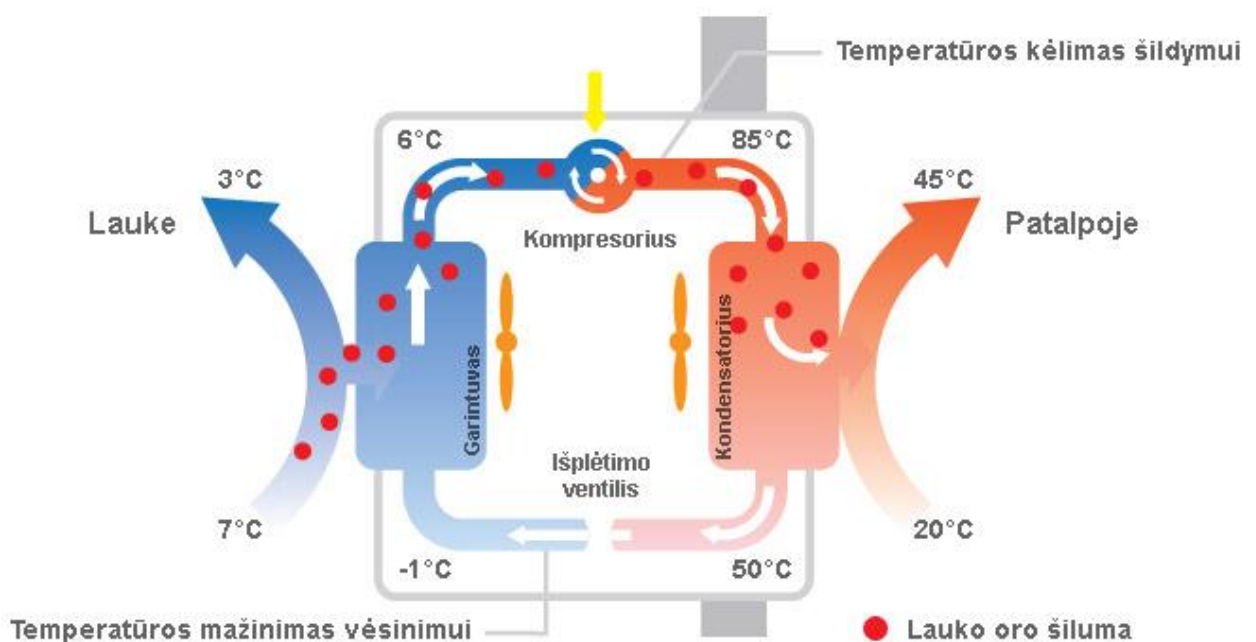
$$G_{min} = (U_L (t_v - t_a)) / (\tau\alpha) \quad (11)$$

1.4. Šilumos siurbliai

1.4.1 Šilumos siurblys ir jo veikimo principas

Šilumos siurblys yra įrenginys, kuris naudojamas šilumos, paimtos iš žemos temperatūros aplinkos (šilumos šaltinio, kuriuo gali būti aplinkos oras, gruntas ir pan.), pernešimui į aukštesnės temperatūros aplinką (aušintuvą, pvz., pastato šildymo ir karšto vandens ruošimo sistemą). Šiam procesui atlikti šilumos siurblyje naudojamas mechaninis darbas arba aukštos temperatūros šilumos šaltinis.

Šilumos siurblių sudaro kompresorius, kondensatorius (šilumokaitis), išsiplėtimo ventilis, garintuvas (1.7 pav.).



1.7 pav. Šilumos siurblio sandara [12]

Atvirkštinis Karno ciklas

Šilumos siurblio darbo ciklas yra atitinka atvirkštinį Karno ciklą.

Tam, kad vyktų bet koks atvirkštinis ciklas, reikia atlikti darbą, nes šiluma iš žemos temperatūros aplinkos turi būti perkeliama į aukštesnės temperatūros aplinką.

Panagrinękime grįžtamąjį Karno ciklą 1234 (1.8 pav.). Izoterminiame plėtimosi procese 2-3 iš žemos temperatūros aplinkos paimama šiluma q_2 . Procese 3-4 darbo kūnas adiabiatiškai suslegiamas, jo temperatūra pakyla nuo T_2 iki T_1 . Toliau slėgimas izotermiškas (4-1), dėl to darbo kūnas atiduoda aplinkai aukštesnės temperatūros šilumą q_1 .

Nuo būsenos 1 iki 2 darbo kūnas adiabiatiškai plečiasi, jo temperatūra nuo T_1 , sumažėja iki T_2 , ir ciklas užsidaro. Ciklo plotas 1234 vaizduoja šilumos kiekį $q_1 - q_2$, ekvivalentišką ciklui atlikti sunaudotam darbui [13].

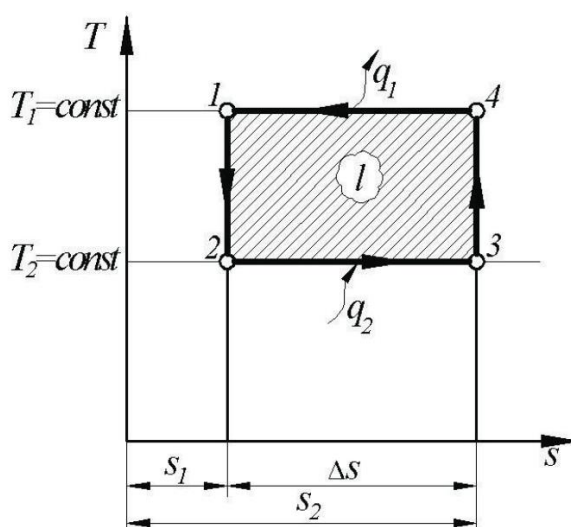
Šaldymo mašinos, dirbančios atvirkštiniu Karno ciklu, šaldymo koeficientas (našumo koeficientas) [13]:

$$COP_{\text{šaldymo}} = Q / W = q_2 / (q_1 - q_2) = T_2 \Delta s / (T_1 \Delta s - T_2 \Delta s) = T_2 / (T_1 - T_2) \quad (12)$$

Čia:

Q – atimtas (atiduotas) šilumos kiekis iš rezervuaro, J;

W – šilumos siurblio kompresoriaus atliktas darbas, J.



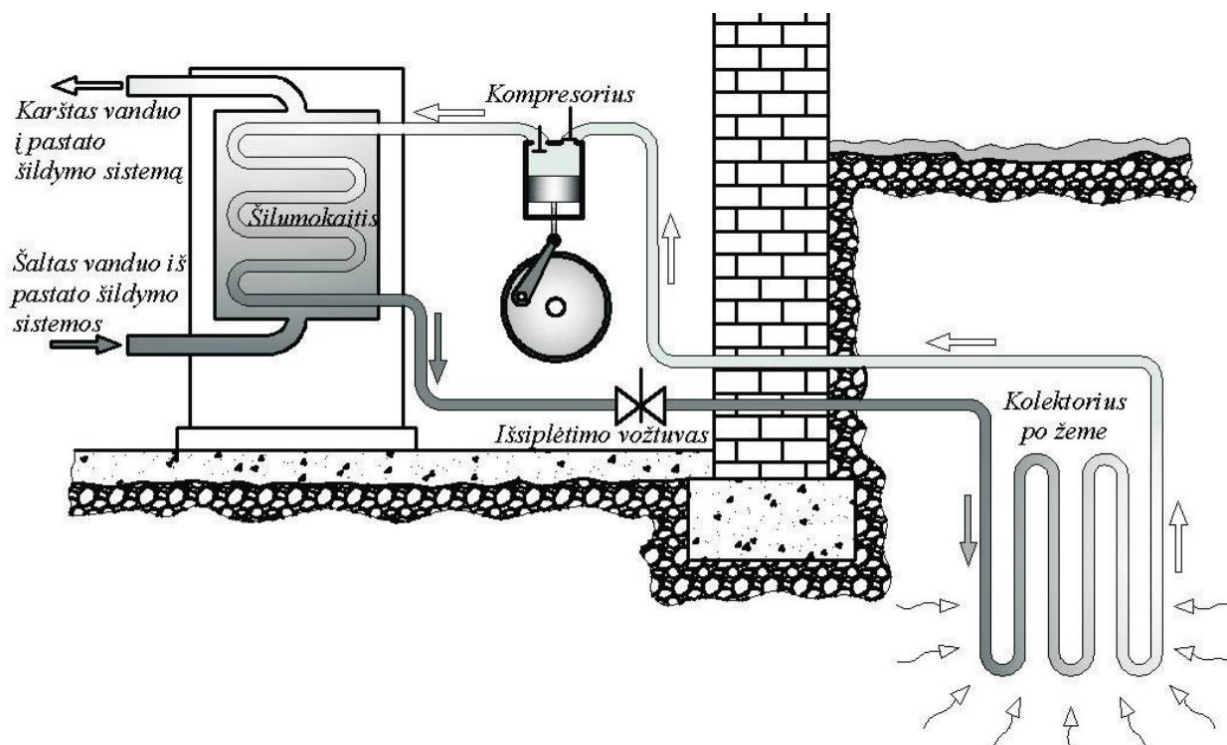
1.8 pav. Atvirkštinis Karno ciklas [13]

Šilumos siurblio, kurio darbas pagrįstas atvirkštiniu Karno ciklu, šildymo koeficientas (našumo koeficientas) [13]:

$$COP_{\text{šildymo}} = Q / W = q_1 / (q_1 - q_2) = T_1 \Delta s / (T_1 \Delta s - T_2 \Delta s) = T_1 / (T_1 - T_2) \quad (13)$$

Be šildymo ir šaldymo ciklų, šilumos siurblyje yra naudojamas ir trečiasis – atitirpinimo ciklas, kuris skirtas spiralių apsaugai nuo pernelyg didelio ledo kiekio susikaupimo. Atitirpinimo ciklas yra būtinas tuo atveju, jei kaip šilumos šaltinis yra naudojamas aplinkos oras, o kaip šilumos šaltinį naudojant gruntą ar gruntinius vandenį jis dažniausiai nėra būtinas.

Šilumos siurblio šildymo ciklas prasideda, kai šaldymo agentas pradeda cirkuliuoti per lauke esančias šilumos siurblio spirales, vadinamas bendru kolektoriaus pavadinimu. Principinė šilumos siurblio veikimo schema šilumos cikle yra pavaizduota 1.9 paveiksle. Šioje schemoje pavaizduotas šilumos siurblys, kurio šilumos šaltinis yra grunte sukauptą šilumą [13].



1.9 pav. Šilumos siurblio veikimo schema šildymo cikle [13]

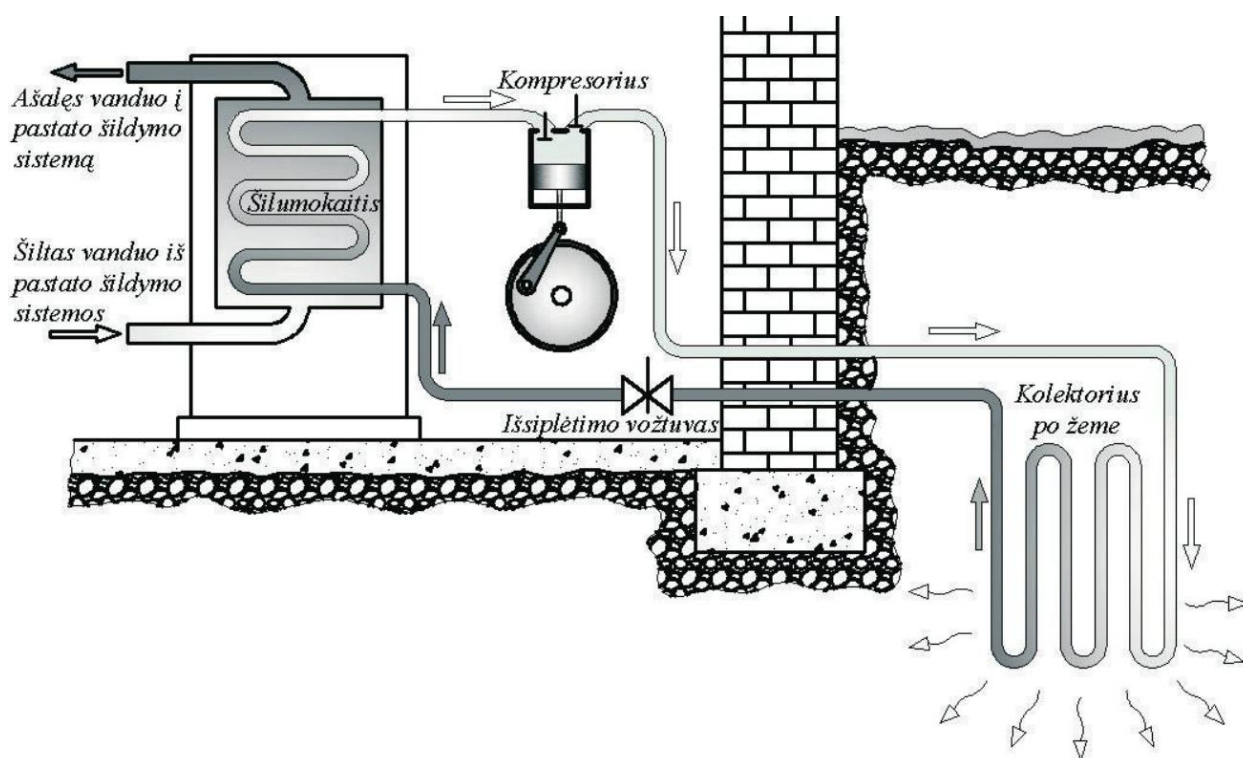
Iš pradžių šaldymo agentas būna žemo slėgio ir žemos temperatūros skystis, bet jis greitai absorbuoja pakankamai šilumos iš aplinkos ir jo temperatūra pakyla iki virimo taško. Tada keičiasi šaldymo agento būvis ir jis virsta karštu garu arba dujomis. Garai suslegiami kompresoriumi ir būdami aukšto slėgio ir aukštos temperatūros prateka spirale, kuri yra pastato viduje esančiame šilumokaityje. Šią spiralę iš išorės apteka vėsus patalpos oras arba šildymo sistemos vanduo, todėl šaldymo agento garai ataušta ir vėl susikondensuoja į skystį. Kondensuodamiesi šaldymo agento garai išskiria slaptąją garavimo šilumą, kuri panaudojama pastatui šildyti arba karštam vandeniui ruošti. Šaldymo agentui vėl virtus skysčiu, jis prateka pro išsiplėtimo vožtuvą ir yra grąžinamas į išorinę spiralę, kur ciklas kartojasi [13].

Reikia atkreipti dėmesį, kad dažniausiai šilumos siurblio sistemoje yra įrengiami du cirkuliacijos kontūrai ir papildomas šilumokaitis. Vienu kontūru, kurį sudaro grunte esantis kolektorius, cirkuliuoja neužšalantis skystis, atiduodantis iš grunto paimtą šilumos energiją per papildomą šilumokaitį pagrindiniam šilumos siurblio kontūrai, kuriame cirkuliuoja šaldymo agentas. Tiesioginė šaldymo agento cirkuliacija požeminiu kolektoriumi yra naudojama retai dėl to, kad tokiu atveju kolektorius turi būti gaminamas ne iš plastiko, bet iš plastiku dengto vario, o pažeidus plastikinę izoliaciją vidinis vario vamzdis greitai pažeidžiamas korozijos ir kontūras prakiūra [13].

Šilumos siurbliui veikiant šaldymo ciklu reversine sklende šaldymo agentas teka priešinga kryptimi, todėl kompresorius pumpuoja šaldymo agentą į kitą pusę nei šildymo cikle, ir

šilumokaityje pastato šilumnešis (oras arba vanduo) yra šaldomas ir vėsina patalpų orą. Taigi šaldymo cikle šaldymo agentas išgaruodamas vidinėje spiralėje (veikiančioje kaip garintuvas) paima šilumą iš patalpų susidarę šaldymo agento garai suslegiami kompresoriumi ir pumpuojami į išorinę spiralę (kuri veikia kaip kondensatorius) ir joje sukondensuojami, kai spiralės išorinis paviršius kontaktuoja su vėsia aplinka (oru, gruntu ar vandeniu) [13]. Kondensacijos metu aplinka sugeria išsiskyrusį šilumos kiekį ir susikondensavęs į skystį šaldymo agentas pro išsiplėtimo vožtuvą gražinamas į pastato vidų. Principinė šilumos siurblio veikimo schema šaldymo cikle pavaizduota 1.10 paveiksle.

Kadangi 1.9 ir 1.10 paveiksluose pavaizduotas šilumos siurblys kaip šilumos šaltinį naudoja grunte sukauptą šilumą jam nėra poreikio veikti trečiuoju - atitirpinimo - ciklu, nes užkasus kolektorių pakankamai giliai grunto temperatūra ištisus metus išlieka pakankamai aukšta, t.y. nenukrenta žemiau 0°C, ir kolektorius apledėti negali [13]. Tačiau kaip šilumos šaltinį naudojant aplinkos orą, esant tam tikrai temperatūrai ir oro drėgmei, išorinė spiralė, kuri šildymo cikle veikia kaip garintuvas, gali pasidengti ledo sluoksniu. Dėl to šilumos siurblys negali tinkamai veikti ir turi būti pašalintas [13].



1.10 pav. Šilumos siurblio veikimo schema šaldymo cikle [13]

Atitirpinimo cikle šilumos siurblio veikimas tam tikru dažnumu keičiamas iš šildymo į šaldymo ciklą. Perėjus į šaldymo ciklą išorinė spiralė yra sušildoma iš patalpos imama šiluma ir

ledo sluoksnis nutirpsta. Sistema išlaikoma šaldymo cikle tol, kol spiralės temperatūra pakyla iki 14 °C. Atitirpinimo laikas priklauso nuo to, kiek ledo būna susikaupę ant spiralės.

1.4.2 Šilumos siurblių tipai

Šilumos siurbLIAI dažnai klasifikuojami pagal jų šilumos šaltinio (šilumnešio) tipą. Trys dažniausiai naudojami šilumos šaltiniai pastatams Šildyti ir karštam vandeniui ruošti yra [13]:

- Oras – vanduo. Pro vieną šilumokaitį cirkuliuoja oras, o pro kitą - skystis (šilumnešis).
- Vanduo – vanduo. Abejuose šilumokaičiuose cirkuliuoja skystas šilumnešis.
- Oras – oras. Pro abu šilumokaičius cirkuliuoja oras.

Šilumos siurblių šaltiniais gali būti:

- Oras – temperatūros intervale nuo 15 °C iki -10 °C.
- Vanduo – temperatūros intervale nuo 15 °C iki 4 °C.
- Gruntas – temperatūros intervale nuo 10 °C iki 0 °C.

Šilumos šaltinis oras

Saulės sušildyto oro yra visur. Šilumos siurblys esant -18 °C temperatūrai dar pakankamai gauna šilumos iš aplinkos. Oro, kaip šilumos šaltinio trūkumas yra tas, kad jis šalčiausias, kai reikalinga šiluma. Nors pavyksta šilumą gauti ir esant -18 °C temperatūrai, tačiau šilumos siurblio išorinė temperatūra sumažėja. Todėl yra naudojama mišri kombinacija su kitu šilumos gamintoju, kuris trumpu (ypač šaltu) metų laiku, padeda šildant patalpas. Ypatingas privalumas - yra paprasta oro šilumos siurblio instaliacija, nes nereikalingi dideli žemės ar šulinio gręžimo darbai.

Šilumos šaltinis vanduo

Grunto vanduo yra gera saulės šilumos saugykla. Netgi šalčiausiomis žiemos dienomis jis išlaiko pastovią nuo +8 °C iki +12 °C temperatūrą. Tai yra privalumas. Remiantis nesikeičiančiu šilumos šaltinio temperatūros lygiu, šilumos siurblio galingumo koeficientas yra tinkamas visus metus. Deja, ne visur yra pakankamai ir reikiamos kokybės gruntinio vandens. Bet ten, kur tai galima, tokie šilumos siurbLIAI apsimoka. Norint panaudoti šilumą, reikia įrengti siurbimo ir išmetimo (vandens nubėgimo) šulinius. Upės ir ežerai taip pat yra tinkami šilumai paimti, nes jie taip pat yra šilumos saugyklos.

Šilumos šaltinis gruntas

Ir šaltomis, ir šiltomis dienomis 1,5 – 1,8 m gylyje žemė lieka pakankamai šilta, kad būtų galima panaudoti šilumos siurbLį. Būtina sąlyga - gana didelio žemės ploto panaudojimas, kad būtų galima nutiesti vamzdyną, kuris surinktų grunto šilumą. Ištraukiamoji kolektoriaus galia lygi: drėgnoje žemėje 30 W/m² ir 10 – 15 W/m² sausoje smėlingoje žemėje. Vamzdžiais teka aplinkai

nekenksmingas mišinys, kuris neužšąla ir paimtą šilumą sunėša į šilumos siurblio aparato kondensatorių. Pagrindinė taisyklė: reikia 3 – 4 kartais daugiau žemės ploto, nei šildomas plotas.

Šilumos siurblių su šilumos šaltiniu gruntas pagrindinis trūkumas įrengimo kaina, be to reikalingas didelis sklypo plotas jam įrengti. Pasyviojo namo šildymui ekonominiais sumetimais labiau tiktų šilumos siurblys, kurio šilumos šaltinis yra oras. Tokio šilumos siurblio privalumas yra kompaktiškumas ir paprasta instaliacija. Be to pasyviajam namui reikalingi mažos galios šilumos siurbliai, kurių pasiūla yra maža, o sumontavus didelės galios šilumos siurblių atsirastų perteklinė energija, kuriai akumuliuoti būtų reikalinga taikyti akumuliacinio būdus.

1.5. Pontriagino minimumo (maksimumo) principas

Pontriagino minimumo (maksimumo) principas yra naudojamas optimaliojoje valdymo teorijoje siekiant surasti tinkamiausią valdymą dinaminės sistemos perkėlimui iš vienos būsenos į kitą, ypatingai esant būsenos ar įėjimo valdymo suvaržymams (pradinėms sąlygoms). 1956 metais šią teoriją kartu su savo studentais suformulavo rusų matematikas Levas Semenovičius Pontriaginas (rus. - Лев Семёнович Понтрягин). Kaip išskirtinis atvejis šioje teorijoje yra panaudota variacinio skaičiavimo Eulerio - Lagranžo lygtis.

Neformaliai principas teigia Hamiltonianas turi būti minimizuojanti valdymo poveikį U – visų leidžiamų valdymų aibė. Jeigu $u^* \in U$ yra optimalus problemos valdymas, tuomet principas teigia, kad [14]:

$$H(x^*(t), u^*(t), \lambda^*(t), t) \leq H(x^*(t), u, \lambda^*(t), t) \quad (14)$$

$\forall u \in U$, kur $x^* \in C [t_0, t_f]$ yra optimali būsenos trajektorija ir $\lambda^* \in B [t_0, t_f]$ yra optimali vektor funkcijos trajektorija.

Rezultatas pirmą kartą sėkmingai buvo pritaikytas minimaliojo laiko problemose su suvaržytu įėjimo signalo valdymu, tačiau jis taip pat gali būti naudingas tiriant suvaržytos būsenos problemose.

Hamiltonianui taip pat gali būti išvestos specialios sąlygos. Kai galutinis laikas t_f yra fiksuotas ir Hamiltonianas nėra visiškai priklausomas nuo laiko ($\partial H / \partial t \equiv 0$), tuomet [14]:

$$H(x^*(t), u^*(t), \lambda^*(t)) \equiv const \quad (15)$$

$$H(x^*(t), u^*(t), \lambda^*(t)) \equiv 0 \quad (16)$$

Kai Pontriagino minimumo principas tenkinamas išilgai trajektorijos, jis tampa būtina sąlyga optimumui. Hamiltono-Jakobo-Belmano lygtis optimumui sudaro būtiną ir pakankamą sąlygą, tačiau ši sąlyga turi būti išpildyta visoje būsenos erdvėje [14].

Jei valdymo objektas yra neautonominis ir jis vaizduojamas diferencine lygtimi [14]:

$$dX/dt = \Psi(X(t), U(t), t), \quad (17)$$

o proceso kokybė vertinama kriterijumi [14]:

$$J = \int_{t_0}^T G(U(t), Y(t), t) dt, \quad (18)$$

tai taikant šį metodą, objekto būsenos vektorius yra išplečiamas, įvedant dvi papildomas koordinates [14]:

$$x_0 = J, \quad (19)$$

$$x_{n+1} = t, \quad (20)$$

Vektorinė judesio lygtis yra papildoma dviem lygtimis [14]:

$$dx_0/dt = G(X, U, t) \quad (21)$$

ir

$$dx_{n+1}/dt = 1. \quad (22)$$

Jei objektas yra autonominis, įvedama tik viena papildoma koordinatė X_0 ir viena papildoma lygtis $dx_0/dt = G(X, U, t)$. Tad metodas yra bendras tiek autonominiams, tiek neautonominiams objektams.

Suformuojamas naujas būsenos vektorius [14]:

$$\mathbf{X}^* = \{x_0, x_1, \dots, x_{n+1}\}^T \quad (23)$$

ir vektorinė judesio lygtis:

$$d\mathbf{X}^*/dt = \Psi(\mathbf{X}(t), U(t)). \quad (24)$$

Įvedama papildoma vektor-funkcija

$$\mathbf{P}(t) = \{p_0, p_1, \dots, p_{n+1}\}^T \quad (25)$$

ir sudaroma *Hamiltono* funkcija (*Hamiltonianas*), kaip skaliarinė šios vektor-funkcijos ir išplėstojo būsenos vektoriaus sandauga [14]:

$$H(\mathbf{P}(t), \mathbf{X}^*(t), U(t)) = \mathbf{P}(t)^T \cdot \Psi(\mathbf{X}^*(t), U(t)) = \sum p_i(t) \cdot \psi_i(\mathbf{X}^*(t), U(t)). \quad (26)$$

Pontriagino maksimumo principas teigia, kad tam jog valdymo poveikis $U(t)$ būtų optimalus, turi egzistuoti tokia vektor-funkcija $\mathbf{P}(t)$, kad visame valdymo intervale H funkcija būtų maksimali.

Optimizavimo uždavinys – nustatyti valdymo poveikio $U(t)$ kitimo dėsnį, suteikiantį Hamiltono funkcijai maksimumą visame valdymo intervale ir apibrėžti vektor-funkcijos elementus. Kiekvienu atveju šis uždavinys sprendžiamas individualiai.

Tarkime, kad pavyko nustatyti valdymo poveikio funkcinę priklausomybę $U(t) = F(\mathbf{P}(t), \mathbf{X}(t), t)$. Toliau reikia nustatyti vektor-funkcijos elementų įeinančių į valdymo poveikio išraišką reikšmes ir įrašyti jas į optimalaus valdymo poveikio išraišką. Jei tai pavyksta padaryti - uždavinys yra išspręstas.

Vektor-funkcijos elementams $p_0(t) \dots p_i(t)$ apibrėžti sudarome kanoninių lygčių sistemą [14]:

$$d\mathbf{P}/dt = - [[\partial H(\mathbf{P}(t), \mathbf{X}^*(t), U(t)) / \partial \mathbf{X}^*]^T. \quad (27)$$

arba skaliarinėje formoje:

$$dp_i/dt = - \partial H(\mathbf{P}(t), \mathbf{X}^*(t), U(t)) / \partial x_i. \quad (28)$$

Integruodami kiekvieną (28) lygčių sistemos lygtį ir įvertindami ribines reikšmes nustatome funkcines $p_0(t) \dots p_i(t)$ elementų priklausomybes. Priklausomai nuo uždavinio sąlygų, funkcija $\mathbf{P}(t)$ įgyja skirtingas ribines reikšmes [14]:

- kai kairysis trajektorijos taškas $\mathbf{X}(t_0)$ ir valdymo laikas $t_0 - T$ yra fiksuoti, o dešinysis trajektorijos taškas $\mathbf{X}(T)$ yra laisvas $\mathbf{P}(T) = (-1, 0, 0, \dots, 0)^T \leq 0$;
- kai abu trajektorijos taškai ir valdymo laikas yra fiksuoti $p_0(t) = \text{const}$;
- kai abu trajektorijos taškai fiksuoti, o valdymo laikas – laisvas $p_0(t) = \text{const} \leq 0$ ir $\max H(\mathbf{P}(T), \mathbf{X}^*(T), U(T)) = 0$;
- kai dešinysis trajektorijos taškas ir valdymo laikas yra laisvi $\mathbf{P}(T) = (-1, 0, 0, \dots, 0)^T$ ir $\max H(\mathbf{P}(T), \mathbf{X}^*(T), U(T)) = 0$.

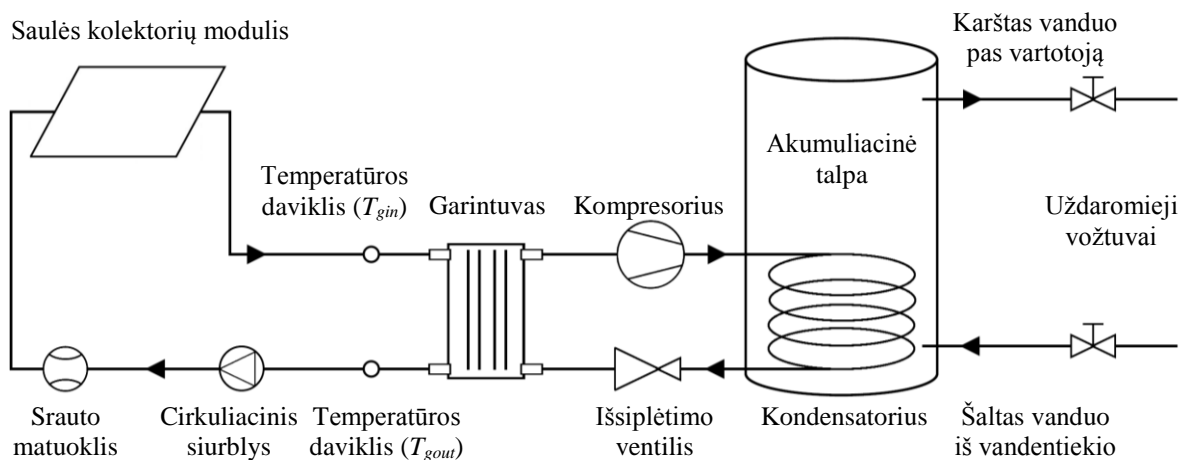
Vektor-funkcijos $\mathbf{P}(t)$ elementai visiškai apibrėžiami integruojant valdymo objekto judesio lygtis ir įvertinant ribines būsenos koordinačių reikšmes [14].

2. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS MODELIO SUDARYMAS

2.1. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos struktūrinė schema

Sukuriama kombinuota saulės kolektorių ir šilumos siurblio sistemos struktūrinė schema, kurią sudaro: saulės kolektorių modulis, garintuvas, kompresorius, kondensatorius, išsiplėtimo ventilis, akumuliacinė talpa, cirkuliacinis siurblys, srauto matuoklis, temperatūros davikliai, uždaramieji vožtuvai, garintuvo ir kondensatoriaus šilumokaičiai.

Šioje sistemoje yra trys šilumos kontūrai: polipropilenglikolio, freono ir vandens. Polipropilenglikolio šilumos kontūras – tai saulės kolektorių modulio kontūras, kuris turi bendrą šilumokaitį su šilumos siurblio garintuvu. Freono šilumos kontūras – tai šilumos siurblio kontūras, kurį sudaro: garintuvas, kompresorius, kondensatorius, išsiplėtimo ventilis. Freono šilumos kontūras turi bendrus šilumokaičius su saulės kolektorių modulio kontūru ir akumuliacinės talpos kontūru. Vandens šilumos kontūras – tai akumuliacinės talpos kontūras, kurį sudaro: akumuliacinė talpa, karšto vandens išleidimo ir šalto vandens paėmimo kontūras.



2.1 pav. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos struktūrinė schema

2.2. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos šiluminis balansas

Visiems sistemos elementams parašomos bendrosios šilumos balanso lygtis.

Kondensatoriaus lygtis:

$$\frac{dT_{cout}}{dt} = \frac{COP E_n}{C_c} - \frac{F_c c_c}{C_c} (T_{cout}(t) - T_{cin}(t)) \quad (29)$$

Čia:

COP – šilumos siurblio naudingumo koeficientas, [-];

E_n – elektrinė šilumos siurblio galia, W;

F_c – kondensatoriaus masės srautas, kg/s;

c_c – kondensatoriaus šilumnešio savitoji šiluma, J/(kg·°C);

$T_{c\text{out}}$ – išeinančio iš kondensatoriaus masės srauto temperatūra, °C;

$T_{c\text{in}}$ – įeinančio į kondensatorių masės srauto temperatūra, °C;

C_c – šiluminė kondensatoriaus talpa, J/°C.

Garintuvo lygtis:

$$\frac{dT_{g\text{out}}}{dt} = \frac{-COP E_n}{C_g} - \frac{F_g c_g}{C_g} (T_{g\text{out}}(t) - T_{g\text{in}}(t)) \quad (30)$$

Čia:

COP – šilumos siurblio naudingumo koeficientas, [-];

E_n – elektrinė šilumos siurblio galia, W;

F_g – garintuvo masės srautas, kg/s;

c_g – garintuvo šilumnešio savitoji šiluma, J/(kg·°C);

$T_{g\text{out}}$ – išeinančio iš garintuvo masės srauto temperatūra, °C;

$T_{g\text{in}}$ – įeinančio į garintuvą masės srauto temperatūra, °C;

C_g – šiluminė garintuvo talpa, J/°C.

Saulės kolektoriaus lygtis:

$$\frac{dT_{s\text{out}}}{dt} = \frac{-F_s c_s}{C_s} (T_{s\text{out}} - T_{\text{sin}}) + \frac{k_1}{C_s} E_s - \frac{k_2}{C_s} \left(\frac{T_{s\text{out}} + T_{\text{sin}}}{2} - T_{\text{laukas}} \right) \quad (31)$$

Arba

$$\frac{dT_{g\text{in}}}{dt} = \frac{-F_s c_s}{C_s} (T_{g\text{in}} - T_{g\text{out}}) + \frac{k_1}{C_s} E_s - \frac{k_2}{C_s} \left(\frac{T_{g\text{in}} + T_{g\text{out}}}{2} - T_{\text{laukas}} \right) \quad (32)$$

Čia:

F_s – saulės kolektoriaus masės srautas, kg/s;

c_s – saulės kolektoriaus šilumnešio savitoji šiluma, J/(kg·°C);

T_{sout} – išeinančio iš saulės kolektoriaus masės srauto temperatūra, °C;

T_{sin} – įeinančio į saulės kolektorių masės srauto temperatūra, °C;

T_{laukas} – lauke esanti temperatūra, °C;

T_{gout} – išeinančio iš garintuvo masės srauto temperatūra, °C;

T_{gin} – įeinančio į garintuvą masės srauto temperatūra, °C;

C_s – šiluminė saulės kolektoriaus talpa, J/°C;

k_1 – empirinis saulės kolektoriaus parametras, m²;

k_2 – empirinis saulės kolektoriaus parametras, W/°C;

E_s – saulės apšvieta, W/m².

Akumuliacinės talpos (boilerio) lygtis:

$$\frac{dT_{cin}}{dt} = \frac{-F_b \cdot c_b}{C_b} (T_{cin} - T_{cout}) - \frac{k_b}{C_b} \left(\frac{T_{cin} + T_{cout}}{2} - T_{patalpos} \right) - \frac{F_{vart} \cdot c_b}{C_b} (T_{cout} - T_{miesto}) \quad (33)$$

Čia:

F_b – boilerio masės srautas, kg/s;

c_b – savitoji boilerio šilumnešio šiluma, J/(kg·°C);

T_{cout} – išeinančio iš kondensatoriaus masės srauto temperatūra, °C;

T_{cin} – įeinančio į kondensatorių masės srauto temperatūra, °C;

$T_{patalpos}$ – patalpos temperatūra, kurioje yra boileris, °C;

T_{miesto} – atitenkančio, iš miesto vandentiekio sistemos, vandens temperatūra, °C;

C_b – šiluminė boilerio talpa, J/°C;

k_b – boilerio empirinis parametras, J/(°C·s);

F_{vart} – vartotojo suvartojamas boilerio šilumnešio masės srautas, kg/s;

Kadangi svarbiausias šio magistrinio darbo tikslas yra surasti optimalų kombinuotos saulės kolektorių ir šilumos siurblio sistemos valdymo metodą, parenkant sistemai optimalios galios šilumos siurblio kompresorių, tad lygčių skaičių sumažiname iki dviejų. Bus nagrinėjamas tik saulės kolektoriaus ir garintuvus, bei su šiais objektais susijusios šilumos balanso lygtis.

Šis supaprastinimas padidins skaičiavimų spartą, sprendinių radimą, palengvins algoritmo užrašymą.

Užrašome lygtį $eq1$, kuri padės surasti atitekančio iš saulės kolektoriaus šilumnešio temperatūrą į garintuvą, kuris yra šilumos siurblyje. Ši temperatūra taip pat yra garintuvo įėjimo temperatūra.

$$eq1 = E_n - COP \cdot (T_{gin}(t) - T_{gout}(t)) \quad (34)$$

$eq1 = 0$, gauname T_{gin} lygtį:

$$T_{gin}(t) = E_n / COP + T_{gout}(t) \quad (35)$$

Iš 35 lygties matyti, kad garintuvo įėjimo ir išėjimo temperatūrų kitimo sparta vienoda, nes $T'_{gin}(t) = T'_{gout}(t)$.

Primame, kad saulės galia E_s yra pastovi nagrinėjamame laiko intervale. Jei nešviečia saulė, tai proceso pradžioje T_{gin} – išeinančio šilumos srauto iš saulės kolektoriaus temperatūra (arba temperatūra garintuvo įėjime) yra lygi lauko (aplinkos) temperatūrai T_{lauko} . Tai atsispindi $eq2$ lygtyje.

$$eq2 = \frac{A \cdot F_m \cdot \left(E_s \cdot \tau\alpha - U_L \cdot \left(\frac{T_{gin}(t) + T_{gout}(t)}{2} - T_{lauko} \right) \right)}{C_s} - \frac{F_s \cdot c}{C_s} (T_{gin}(t) - T_{lauko}) - T'_{gout}(t) \quad (36)$$

Čia:

F_s – saulės kolektoriaus šilumnešio masės srautas, kg/s;

F_m – saulės kolektoriaus naudingumo koeficientas, kuris priklauso nuo skysčio debito ir kanalo konstrukcijos, [-];

c – savitoji kolektoriaus šilumnešio šiluma, J/(kg·°C);

C_s – saulės kolektoriaus ir šilumokaičio visuminė šiluminė talpa, J/°C;

$\tau\alpha$ – kolektoriaus optinio efektyvumo koeficientas, [-];

U_L – suminis šilumos nuostolių koeficientas, perskaičiuotas į skaidrios dangos plotą A (W/(m²·°C))

A – saulės kolektoriaus efektyvus plotas, m²

Pradėjus šviesti saulei per tam tikrą laiką, kuris priklauso nuo inercijos C_s , šilumnešio į šilumos siurblių temperatūra pakyla iki tam tikros T_{gin} nusistovėjusios vertės.

Kai nusistovi stacionarus režimas sistemos nuostolių galia yra lygi generuojamai galiai, kuri priklauso nuo saulės šiluminio poveikio.

Pirmojoje lygtyje $eq1$ darome prielaidą, kad šilumos siurblio laiko pastovioji yra daug kartų mažesnė už antrosios lygties $eq2$ laiko pastoviąją, todėl inercijos dedamosios $eq1$ lygtyje nevertiname.

Pasinaudodami „Wolfram Mathematica 10“ programa (žr. 1 priedą) į $eq2$ lygtį įstatome $eq1$ lygtį ir $T'_{gin}(t)$ išvestinę. Gauname $T_{gout}(t)$ išraišką:

$$T_{gout}(t) = \frac{e^{-\frac{t(cFs+AFmUL)}{Cs}} \left(2cFs \left(En - COPT_{lauko} + e^{\frac{t(cFs+AFmUL)}{Cs}} (-En + COPT_{lauko}) + COPT_p \right) \right)}{2COP(cFs+AFmUL)} + \frac{e^{-\frac{t(cFs+AFmUL)}{Cs}} \left(AFm \left(EnUL - 2COP(Esta + (T_{lauko} - T_p)UL) + e^{\frac{t(cFs+AFmUL)}{Cs}} (-EnUL + 2COP(Esta + T_{lauko}UL)) \right) \right)}{2COP(cFs+AFmUL)} \quad (37)$$

2.3. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos optimizavimo kriterijaus pasirinkimas

Darome prielaidą, kad generuojama galia $eq1$ lygtyje yra įeinančio į šilumos siurblių (garintuvą) ir išeinančio iš šilumos siurblio šilumnešio temperatūros skirtumas $T_{gin}(t) - T_{gout}(t)$.

Pasinaudodami šiuo optimizavimo kriterijumi siekiame išlaikyti pastovią tam tikrą vidutinę išeinančio šilumnešio $T_{gout}(t)$ temperatūrą tam tikrame lygyje T_v , t.y. tam tikrą pageidaujamą išeinančio šilumnešio vidutinę temperatūrą visos paros metu. To reikia, kad būtų sudaromas optimalus temperatūrinis skirtumas tarp saulės kolektoriaus ir garintuvo darbinių temperatūrų. Šis temperatūrinis skirtumas padeda efektyviau paimti saulės kolektoriaus generuojamą energiją ir ją perkelti į akumuliacinę talpą, kur šią energiją gali panaudoti šilumos vartotojas.

Pasinaudodame žiniomis apie Pontriagino minimumo (maksimumo) principą, taip pat nežinomų parametrų įvertinimą ir jų suradimą naudojantis didžiausios santykinės entropijos principus (angl. Parameter Estimation Using Maximum Relative Entropy) [15,16], anksčiau paminėtas optimizavimo kriterijų pavadinkime integraliniu kvadratinu nuokrypiu – H . H - tai apibrėžtinis integralas, kurio laiko intervalo pradžia yra 0, o pabaiga yra Δt .

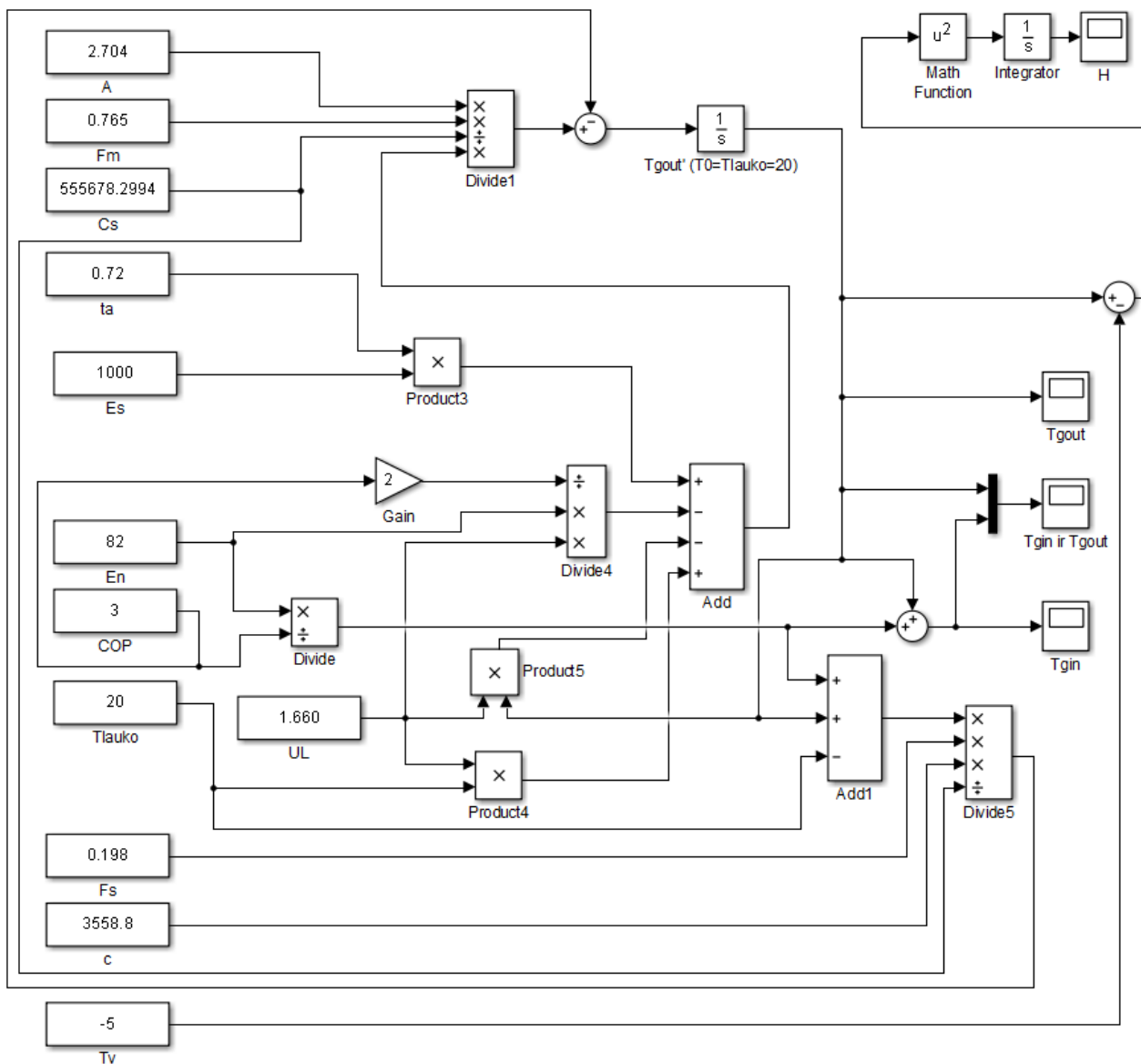
$$H = \int_0^{\Delta t} (T_{gout}(t) - T_v)^2 dt \quad (38)$$

Šis ekstremumas gautas aproksimavus Hamiltoniano integralą stačiakampio metodu [17]. Šis metodas dar vadinamas užauginimo metodu, kuris buvo sukurtas iškiliosios funkcijos (nagrinėjamu atveju gauname parabolę) optimizavimo principais ir santykinės entropijos maksimizacija [15,16]. Įstačius šią formulę į „Wolfram Mathematica 10“ programą išreiškiamas H per jau rastą $T_{gout}(t)$ išraišką gauname:

$$\begin{aligned}
H = & \frac{1}{8COP^2(cFs+AFmUL)^3} (Cs(2cFs(En + COP(-Tlauko + Tp)) + AFm(EnUL - \\
& 2COP(Esta + TlaukoUL - TpUL)))^2 - Cse^{-\frac{2(cFs+AFmUL)\Delta t}{cs}} (2cFs(En + COP(-Tlauko + \\
& Tp)) + AFm(EnUL - 2COP(Esta + TlaukoUL - TpUL)))^2 - 4Cs(2cFs(En + \\
& COP(-Tlauko + Tp)) + AFm(EnUL - 2COP(Esta + TlaukoUL - TpUL))) (2cFs(En + \\
& COP(-Tlauko + Tv)) + AFm(EnUL - 2COP(Esta + TlaukoUL - TvUL))) + \\
& 4Cse^{-\frac{(cFs+AFmUL)\Delta t}{cs}} (2cFs(En + COP(-Tlauko + Tp)) + AFm(EnUL - 2COP(Esta + \\
& TlaukoUL - TpUL))) (2cFs(En + COP(-Tlauko + Tv)) + AFm(EnUL - 2COP(Esta + \\
& TlaukoUL - TvUL))) + 2(cFs + AFmUL)(2cFs(En + COP(-Tlauko + Tv)) + \\
& AFm(EnUL - 2COP(Esta + TlaukoUL - TvUL)))^2 \Delta t)
\end{aligned} \tag{39}$$

2.4. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos „Matlab Simulink“ modelio sudarymas

Sudaromas „Matlab Simulink“ modelis pagal ankstesniuose poskyriuose išvestas lygtis. Šis modelis bus naudojamas lygčių ir įvestų parametų teisingumo patikrinimui.



2.2 pav. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos „Matlab Simulink“ modelis

3. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS ALGORITMO IR PROGRAMOS SUKŪRIMAS

Šiame skyriuje pateikiamas kombinuotos saulės kolektorių ir šilumos siurblio sistemos algoritmas pagal, kurį bus atliekamas nagrinėjamos sistemos modeliavimas, esant skirtingoms aplinkos sąlygos, sistemos parametrams.

Visi algoritmai įgyvendinami kompiuteriu naudojant „Microsoft Visual Studio 2013“ programų paketą ir C# programavimo kalbą.

3.1. Kompresoriaus galia pastovi, režimas „Įjungta/Išjungta“, laiko intervalo kitimas kas dt

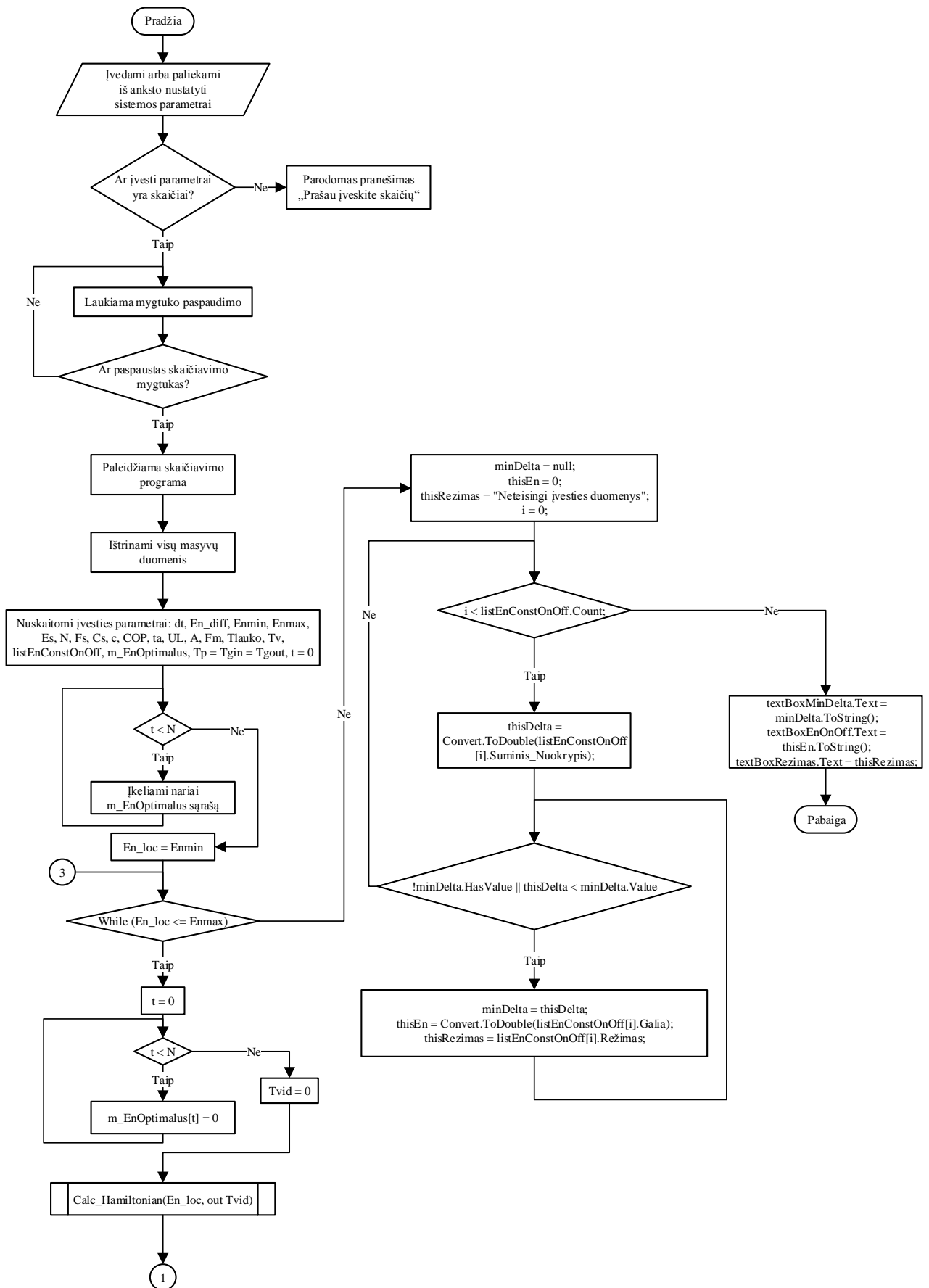
Pirmasis uždavinys, kurį reikia padaryti prieš rašant programą – susiplanuoti, kaip bus realizuojamas programos algoritmas.

Iš ankstesnio skyriaus, jau žinome visos sistemos šilumos balanso lygtis. Taip pat žinome mūsų pagrindinį uždavinį, t.y. surasti optimalią šilumos siurblio kompresoriaus variklio galią En , su kuria integralinis kvadratinis nuokrypis H visame laiko intervale yra mažiausias.

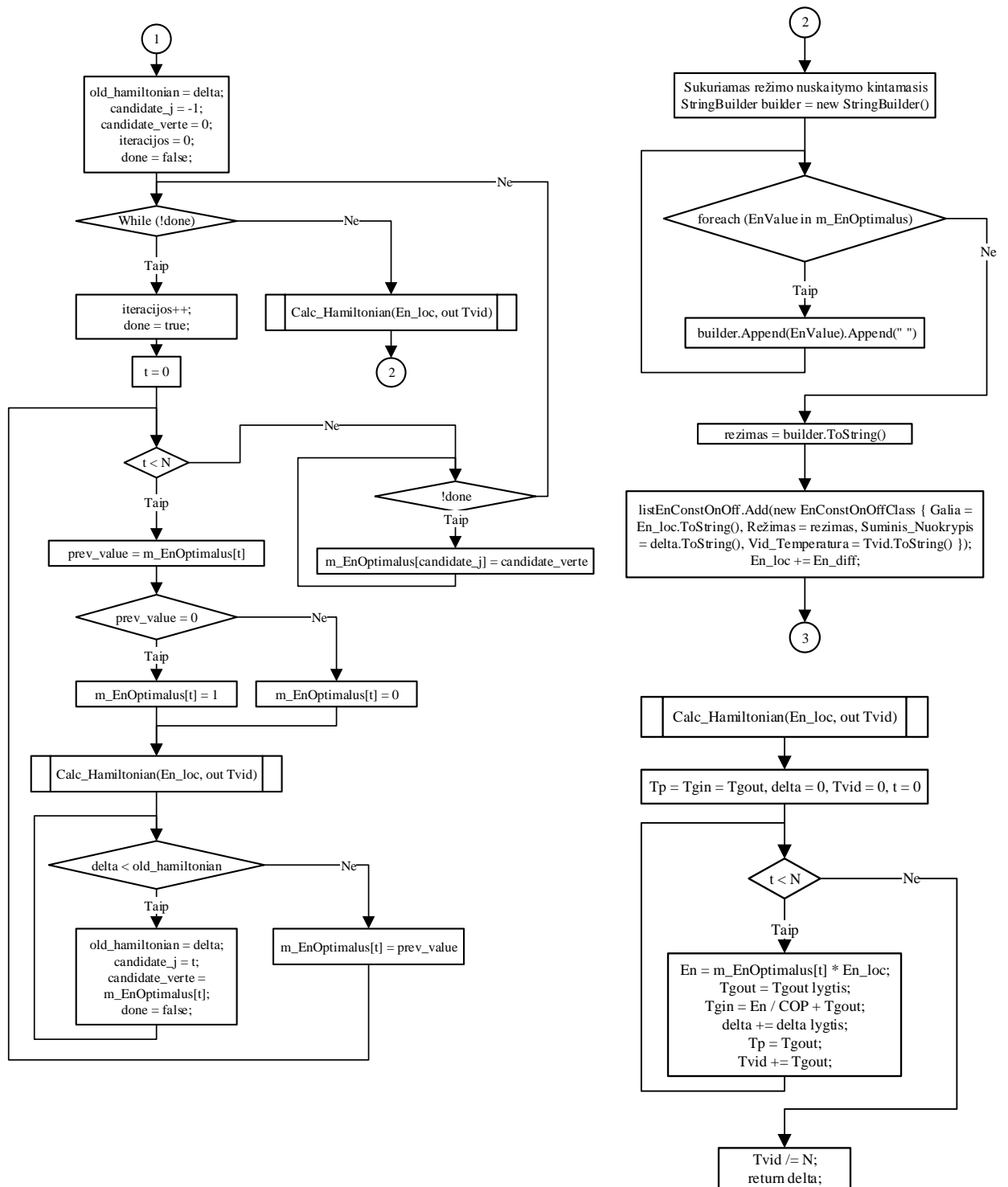
Laiko intervalą nusako laiko žingsnio dydis dt ir laiko žingsnių skaičius N . Algoritme bus ieškoma optimali En galia iš intervalo ($En_{max} - En_{min}$). Čia En_{max} – tai maksimali galima variklio galia, o En_{min} – minimali galima variklio galia. Optimalios variklio galios bus ieškoma didinant variklio galios žingsnį En_{dif} . Variklio galia tikrinama, kai jis yra „įjungtas“ ir „išjungtas“ kiekvienu nurodytu laiko momentu. Renkama optimaliausia „įjungtų“ ir „išjungtų“ variklių kombinacija visame laiko intervale, su kuria suminis kvadratinis nuokrypis yra mažiausias.

Operacijai „Kompresoriaus galia pastovi, režimas „Įjungta/Išjungta“, laiko intervalo kitimas kas dt “ nubraižoma algoritmo veikimo schema, kuri pateikta paveiksluose: 3.1, 3.2.

Svarbi pastaba – algoritmo veikimo schemoje ir sukurtoje programoje optimizavimo kriterijus $H = \text{delta}$.



3.1 pav. Kompresoriaus galia pastovi, režimas „Įjungta/Išjungta“, laiko intervalo kitimas kas dt .
Programos algoritmo duomenų įvedimas ir išvedimas.

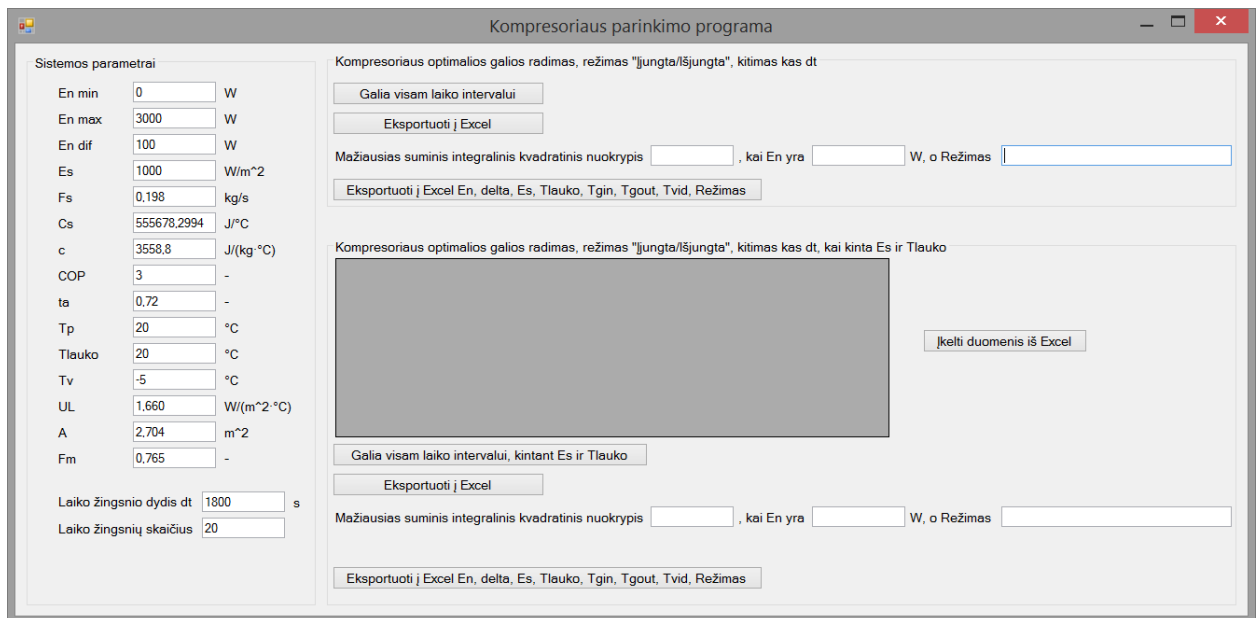


3.2 pav. Kompresoriaus galia pastovi, režimas „Įjungta/Išjungta“, laiko intervalo kitimas kas dt.

Programos pirmo ir antro punktų, bei optimizavimo kriterijaus $H = \text{Calc_Hamiltonian}$ apskaičiavimas.

Pagal pateiktas algoritmo schemas parašoma programa C# kalba. Sukuriama grafinė vartotojo sąsaja, kurioje yra mygtukai skaičiavimų atlikimui, įvesties ir išvesties laukeliai, skirti patogiam parametrų įvedimui rezultatų parodymui. Taip pat sukuriamos papildomos funkcijos, tokios kaip duomenų įkėlimas bei iškėlimas į „Microsoft Excel“ dokumentą.

Parašomas papildomas algoritmas, kuris nuskaito „Microsoft Excel“ dokumente esančias saulės apšvietos E_s ir lauko temperatūros T_{lauko} reikšmes. Įvedus tinkamą laiko žingsnių skaičių N ir laiko žingsnio dydį dt , galima surasti optimaliausią šilumos siurblio variklio valdymo režimą prie besikeičiančių aplinkos sąlygų. Pagrindinis programos langas pateikiamas 3.3 paveiksle.



3.3 pav. Kompresoriaus parinkimo programos langas

4. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS MODELIO ELEMENTŲ PRADINIŲ PARAMETRŲ PASIRINKIMAS IR APSKAIČIAVIMAS

4.1. Saulės kolektoriaus parinkimas

Lietuvoje iš visų saulės kolektorių sistemų pati efektyviausia sistema, kurioje naudojami vakuuminiai saulės kolektoriai, nes jie efektyviau dirba žiemą ir labiau tinkami naudoti šaltesnio klimato juostose, kur saulės kiekis yra mažesnis nei pietuose. Taip pat šie kolektoriai pasirenkami, nes jų efektyvumas gali būti net iki 98 %.

Parašyti programai išbandyti pasirenkamas gamintojo „Sunshore Solar Germany Ltd.“ vakuuminis saulės kolektorius „Q-B-J-90/3.9/0“. Šio kolektoriaus pagrindiniai parametrai pateikti 4.1 lentelėje.

4.1 lentelė. Vakuuminio saulės kolektoriaus QBJ-90/3.90/0 charakteristikos [18]

Paskirtis ir panaudojimas	Karšto vandens gamyba vienbučiuose ir daugiabučiuose gyvenamuosiuose pastatuose, visuomeniniuose pastatuose ar mokyklose, biurų pastatuose, baseinuose, komerciniuose objektuose, šiluminėse jėgainėse
Modelis/Tipas	QBJ-90 / 3.9 / 0
Paviršiaus plotas	4,94 m ²
Efektyvus plotas	2,704 m ²
Kolbų skaičius	30 kolbų
Kolbų tipas/matmenys	58*1800mm
Kolbų medžiaga	grūdintas 3.3 Borosilikatinis stiklas, atlaikantis iki 25 mm dydžio krušą
Šiluminė galia (kWh)	Iki 2700 kWh (Vokietijos duomenys)
Priedai	Rėmas ir tvirtinimo elementai
Galimybės jungti	Į vieną sistemą galima jungti iki 50 kolektorių.
Eksploatavimo sąlygos	Beslėgė sistema (iki 1 atm.) su tiesioginiu šilumos perdavimu šilumnešiu, galimybė naudoti vandenį (sistemoms su reguliatoriais užtikrinančiais apsaugos nuo užšalimo funkciją – pašildymas tenu iki reikiamos temperatūros).
Kolektoriaus svoris	83 kg (tuščio)
Kolektoriaus tūris	89 litrai
Matmenys	2650 x 1980mm (plokštumoje) – (plotis x aukštis).
Matmenys (ant žemės)	2620 x 1460 x 1360 mm (su stovu ant horizontalaus paviršiaus) – (plotis x aukštis x gylis).
Konversijos faktorius	0,7504
Šilumos perdavimo koeficientas a1	1,613 [W/(m ² ·°C)]
Šilumos perdavimo koeficientas a2	0,000 [W/(m ² ·°C)] (nuostoliai į aplinką)
Efektyvi šiluminė talpa c _{eff}	164,4 [kJ / (m ² ·°C)]
Maksimali temperatūra	< 230 °C (nesant šilumnešio cirkuliacijos) reali temperatūra – skysčio virimo temperatūra
Efektyvi šiluminė talpa	870 kWh/m ² (bandymų sąlygomis Freiburgo platumoje)

4.2. Saulės kolektoriaus šilumnešio parinkimas

Saulės kolektorius turi pajėgti dirbti plačiame temperatūriniame diapazone -40°C iki 200°C , tad šilumnešis turi neužšalti ir neužvirti tokiose pačiose ribose, o taip pat gerai pernešti ir išlaikyti šilumą nuo saulės kolektoriaus iki šilumos kaupimo vietos (akumuliacinės talpos). Dėl šios priežasties standartiniame saulės kolektorių kontūre, kaip šilumnešis yra naudojamas polipropilenglikolio ir vandens mišinys, kurio santykis yra 50/50 arba 60/40 [19]. Taip pat šis mišinys pasižymi geromis antikorozinėmis savybėmis.

Šiame magistriniame darbe modeliuojamai sistemai kaip šilumnešį naudosime polipropilenglikolio ir vandens mišinį, kurio santykis yra 50/50. Sistemos parametrų parinkimui yra svarbiausi šie šilumnešio parametrai: savitoji šilumnešio šiluma $c = 3558,78 \text{ J}/(\text{kg}\cdot^{\circ}\text{C})$ ir šilumnešio tankis $\rho = 1041 \text{ kg}/\text{m}^3$ [20].

Standartiškai šilumokaityje šilumnešis juda palyginus nedideliu greičiu. Priimame, kad mūsų sistemoje šilumnešio greitis $F_s = 200 \text{ l/h} = 0,19771 \text{ kg/s}$, kai šilumokaityje teka polipropilenglikolio ir vandens mišinys, kurio santykis yra 50/50.

4.3. Saulės kolektoriaus ir jo šilumos kontūro, kuriame teka polipropilenglikolio ir vandens mišinys, visuminės šiluminės talpos apskaičiavimas

Visuminė šiluminė talpa modeliuojamai sistemai apskaičiuojama pagal 40 formulę:

$$C_s = C_{\text{vamzdyno}} + C_{\text{kolektoriaus}} = c \cdot \rho \cdot V + c_{\text{eff}} \cdot A \quad (40)$$

Čia:

V – šilumokaičio tūris, m^3 . Priimame, kad šilumokaičio vamzdyno tūris $V = 30 \text{ l} = 0,03 \text{ m}^3$.

c_{eff} - efektyvi pasirinktojo saulės kolektoriaus šiluminė talpa, $164400 \text{ J}/(\text{m}^2\cdot^{\circ}\text{C})$. Paimta reikšmė iš 4.1 lentelės.

A – Saulės kolektoriaus efektyvus plotas, $2,704 \text{ m}^2$. Paimta reikšmė iš 4.1 lentelės.

c - savitoji šilumnešio (polipropilenglikolio ir vandens mišinio) šiluma, $3558,78 \text{ J}/(\text{kg}\cdot^{\circ}\text{C})$.

ρ - šilumnešio (polipropilenglikolio ir vandens mišinio) tankis, $1041 \text{ kg}/\text{m}^3$.

Išstatę reikšmes gauname visuminę šiluminę talpą modeliuojamai sistemai :

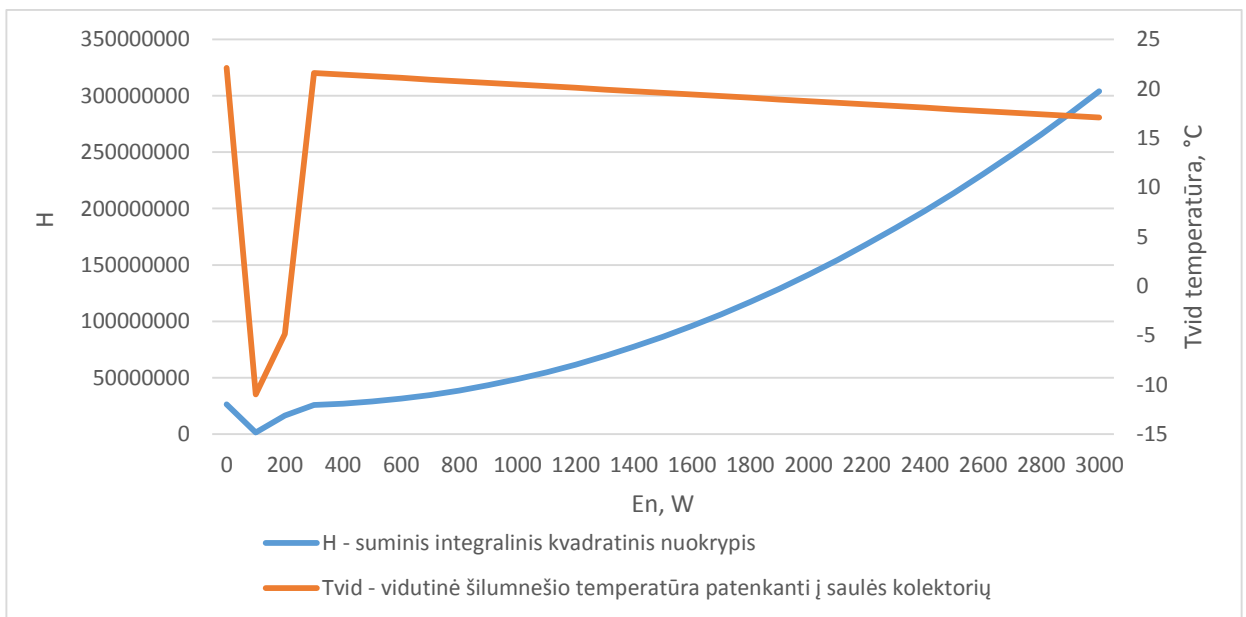
$$C_s = 3558,78 \cdot 1041 \cdot 0,03 + 164400 \cdot 2,704 = 555678,2994 \text{ J}/^{\circ}\text{C}.$$

5. KOMBINUOTOS SAULĖS KOLEKTORIŲ IR ŠILUMOS SIURBLIO SISTEMOS MODELIO EKSPERIMENTINIS TYRIMAS

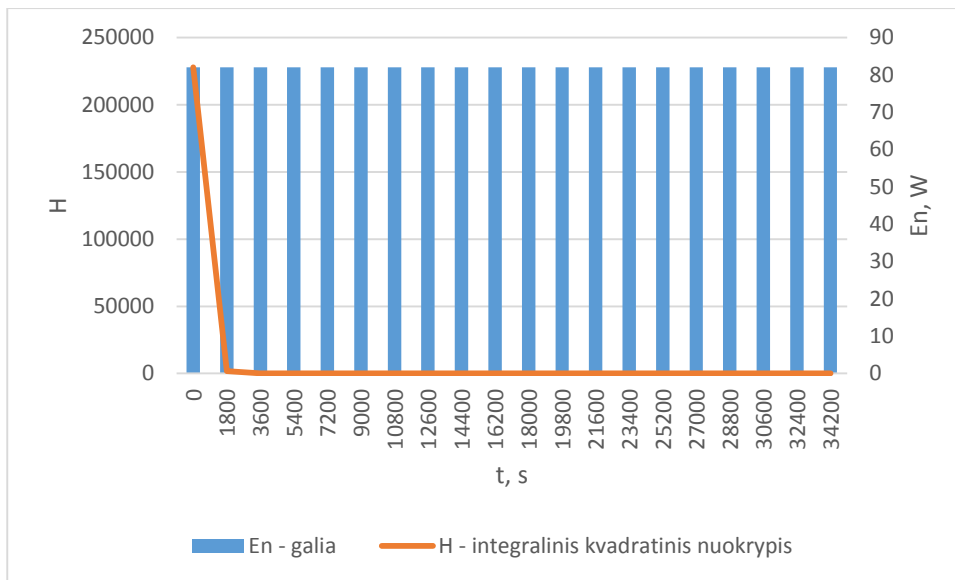
5.1. Optimalios galios kompresoriaus suradimas pasirinktai kombinuotajai sistemai

Pasirinkta kombinuota sistema modeliuojama 10 h (36000 s), kai kompresorius įjungiamas arba išjungiamas kas 0,5 h (3600 s).

Pradiniai sistemos parametrai ir gautasis grafikas pateikti 5.1 paveiksle.

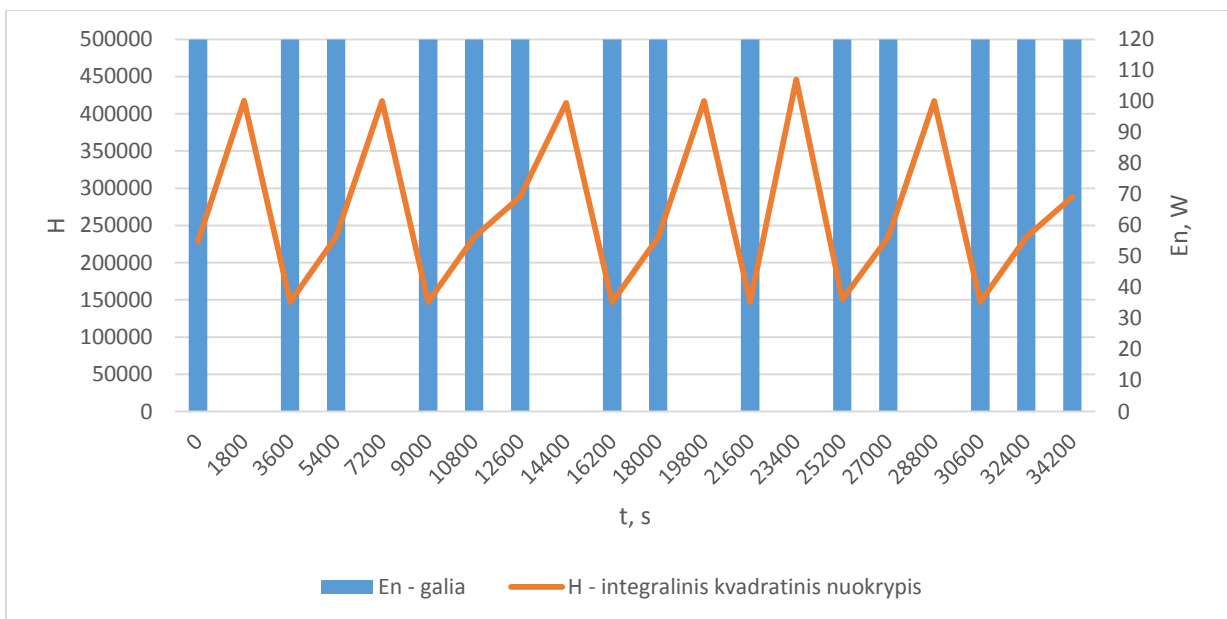


5.1 pav. Kompresoriaus parinkimo programos langas, kai $En_{max} = 3000$ W, $En_{dif} = 100$ W



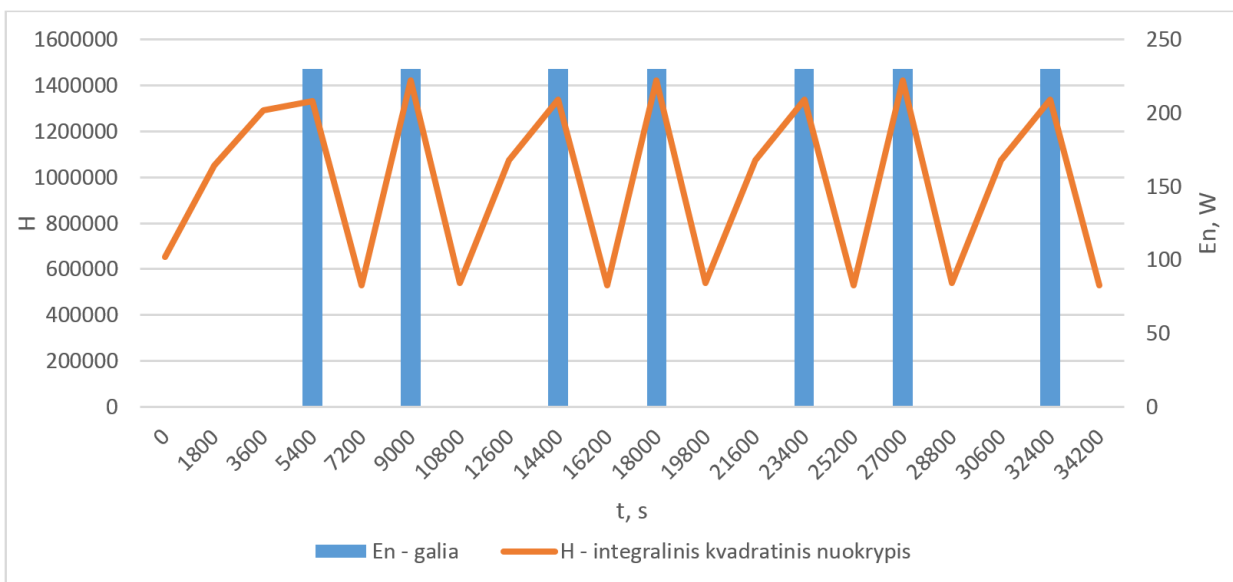
5.4 pav. Kompresoriaus darbo režimas, kai $En = 82$ W

Kai $En = 120$ W, tai kompresorius optimaliausiai veikia režimu (1 0 1 1 0 1 1 1 0 1 1 0 1 0 1 1 0 1 1). Kompresoriaus galios ir integralinio kvadratinio nuokrypio kitimo grafikas nuo laiko pateikiamas 5.5 paveiksle.



5.5 pav. Kompresoriaus darbo režimas, kai $En = 120$ W

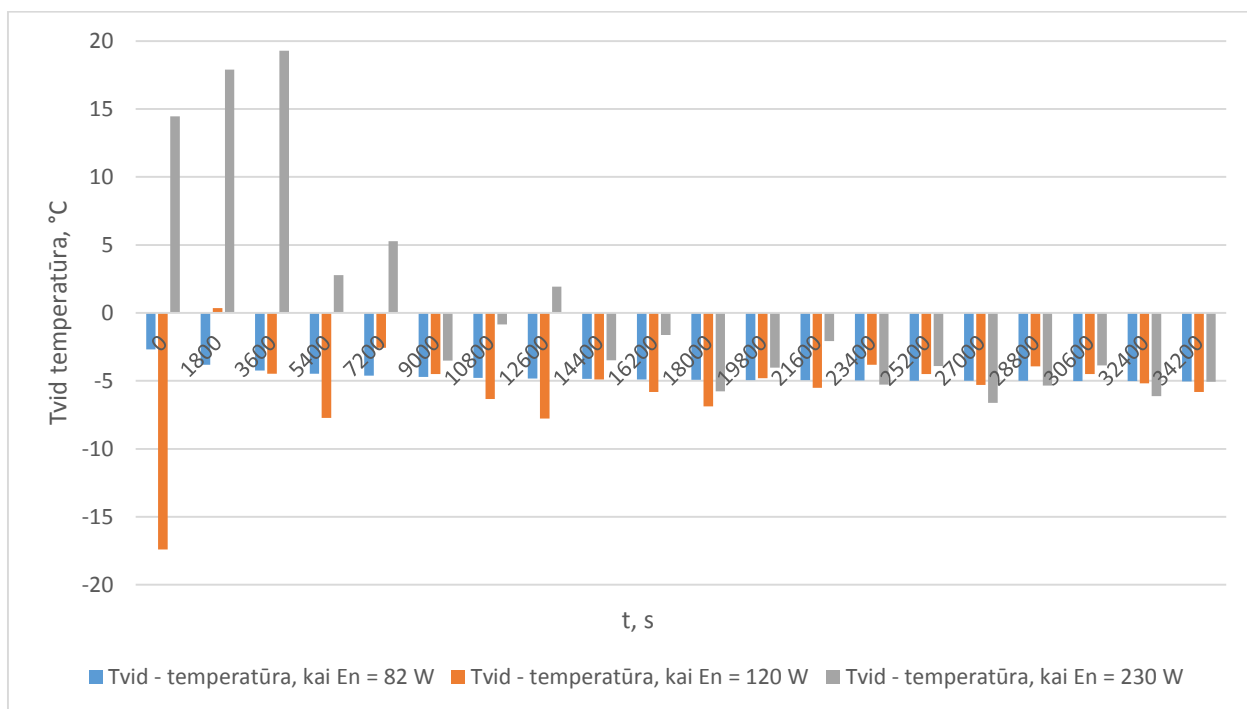
Kai $En = 230$ W, tai kompresorius optimaliausiai veikia režimu (0 0 0 1 0 1 0 0 1 0 1 0 0 1 0 0 1 0 1 0 0 1 0). Kompresoriaus galios ir integralinio kvadratinio nuokrypio kitimo grafikas nuo laiko pateikiamas 5.6 paveiksle.



5.6 pav. Kompresoriaus darbo režimas, kai $En = 230$ W

Išanalizavus 5.4, 5.5 ir 5.6 paveikslus matyti, kad mažiausias integralinis kvadratinis nuokrypis H gaunamas tada, kai $En = 82$ W ir kompresorius dirba nuolat.

5.3. Šilumnešio atitekančio į saulės kolektorių vidutinės temperatūros kaita per 10 h, esant skirtingoms En vertėms

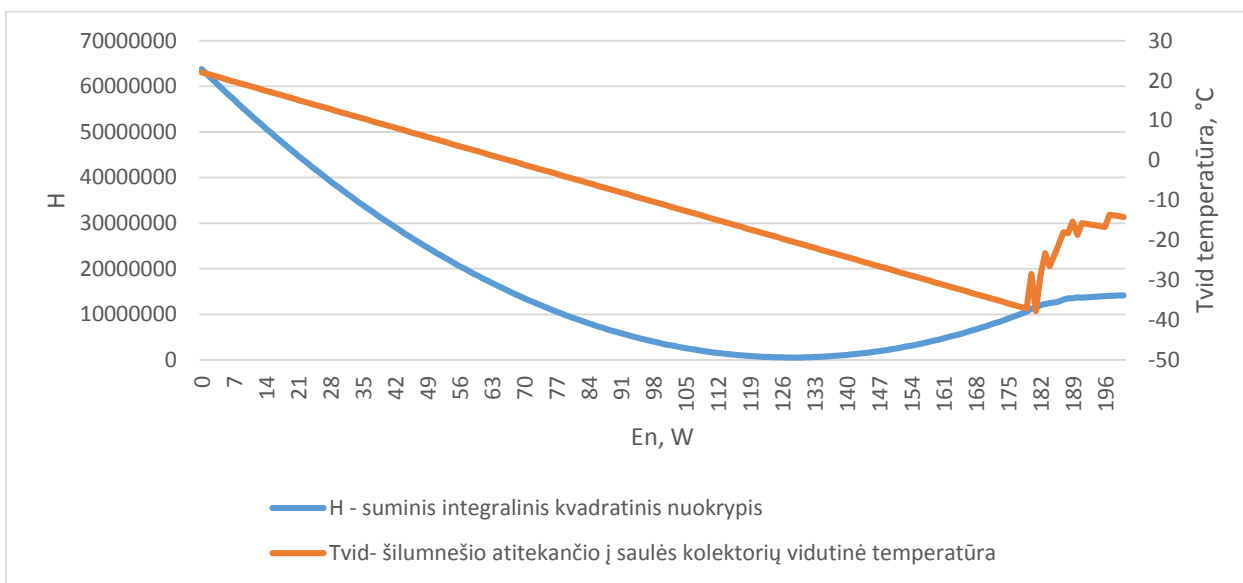


5.7 pav. Šilumnešio atitekančio į saulės kolektorių vidutinės temperatūros kaita per 10 h, esant skirtingoms En vertėms

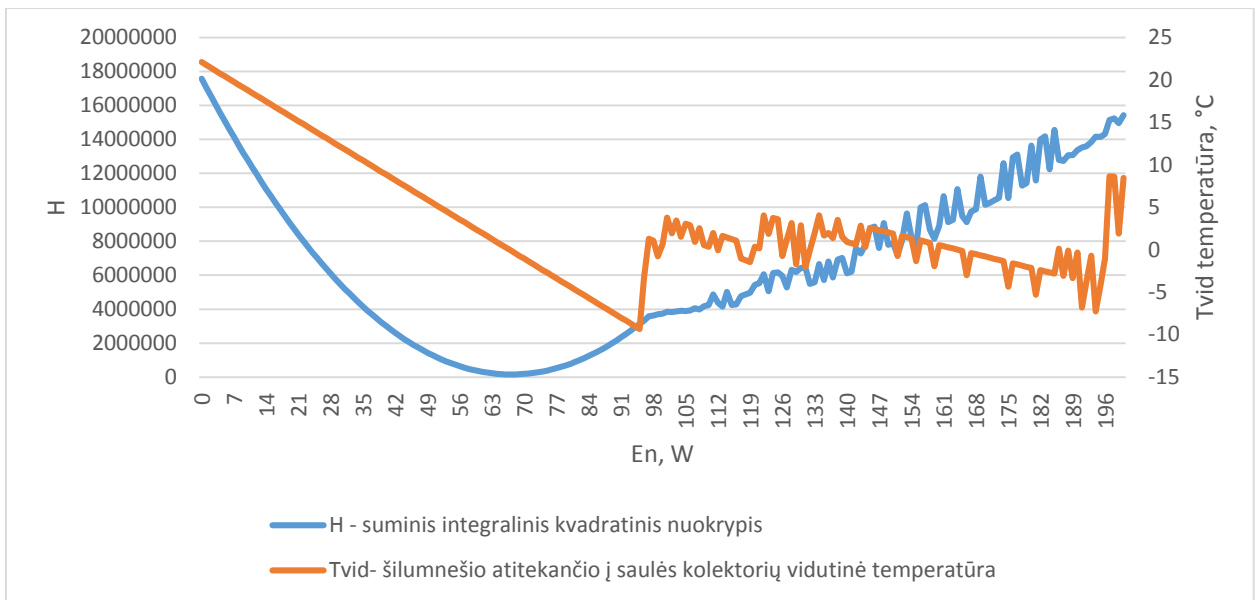
Iš 5.7 paveikslo ir iš anksčiau pateiktų grafikų matyti, kad programa skaičiuoja kokybiškai ir tiksliai. Esant didžiausiam šilumnešio temperatūros T_{vid} nuokrypiui nuo nustatytos programoje T_v reikšmės gauname ir didžiausius H nuokrypius tame laiko intervale.

5.4. Kompresoriaus darbo režimų palyginimas, kai sistema modeliuojama 10 h, o kompresorius gali būti įjungiamas arba išjungiamas kas 0,5 h, taip pat keičiant T_v reikšmę

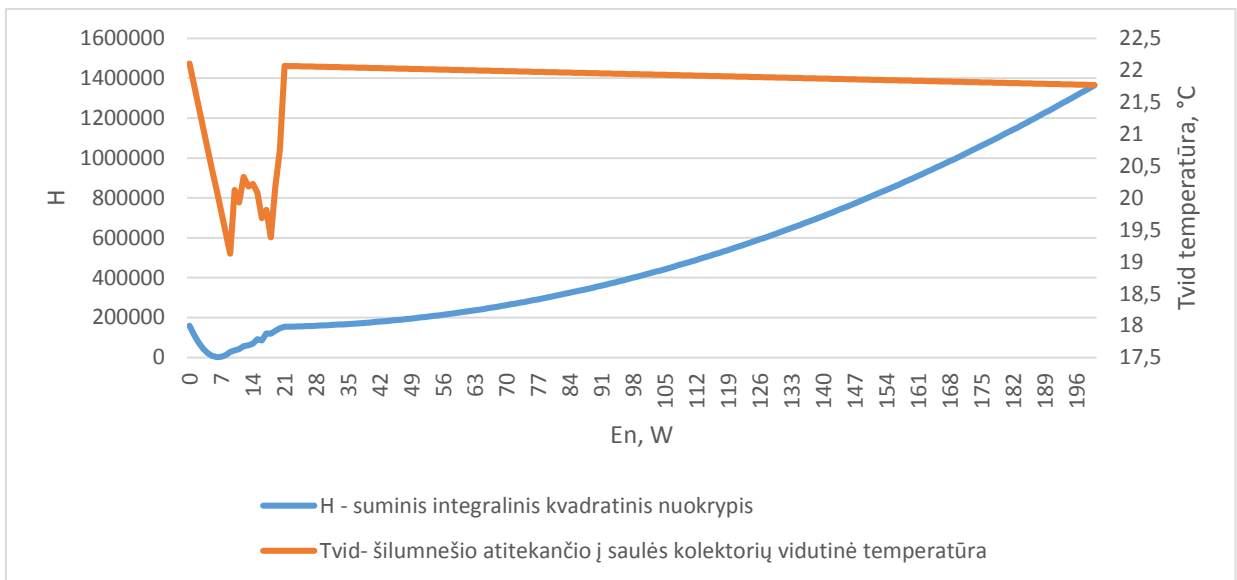
Patikrinkime T_{vid} ir H pokytį, kai keičiama užduotosios temperatūros T_v vertė. Paimame tris T_v reikšmes: $T_v = -20\text{ }^\circ\text{C}$, $T_v = 0\text{ }^\circ\text{C}$, $T_v = 20\text{ }^\circ\text{C}$. Surandame nurodytai temperatūrai optimalios galios kompresoriaus galią ir optimaliausią veikimo režimą.



5.8 pav. T_{vid} ir H pokytis, kai optimalus $En = 128\text{ W}$, gaunamas darbinis režimas „visada įjungta“, kai $T_v = -20\text{ }^\circ\text{C}$



5.9 pav. T_{vid} ir H pokytis, kai optimalus $En = 67$ W, gaunamas darbinis režimas „visada įjungta“, kai $T_v = 0$ $^{\circ}C$



5.10 pav. T_{vid} ir H pokytis, kai optimalus $En = 6$ W, gaunamas darbinis režimas „visada įjungta“, kai $T_v = 20$

Iš gautųjų grafikų ir programos išvestų rezultatų matyti, kad kintant T_v diapazone nuo 20 iki 20, optimalaus kompresoriaus režimas nepasikeitė. Keitėsi tik kompresoriaus galios vertė En . H pokyčiai išliko panašiam lygyje. Jei sistemos darbo metu keisis tik T_v reikšmė, tai galima teigti, kad šiai sistemai tinka kompresorius, kurio galia gali būti reguliuojama diapazone 6 -128 W. Tai ir bus optimaliausias ir efektyviausias kompresorius šiai sistemai.

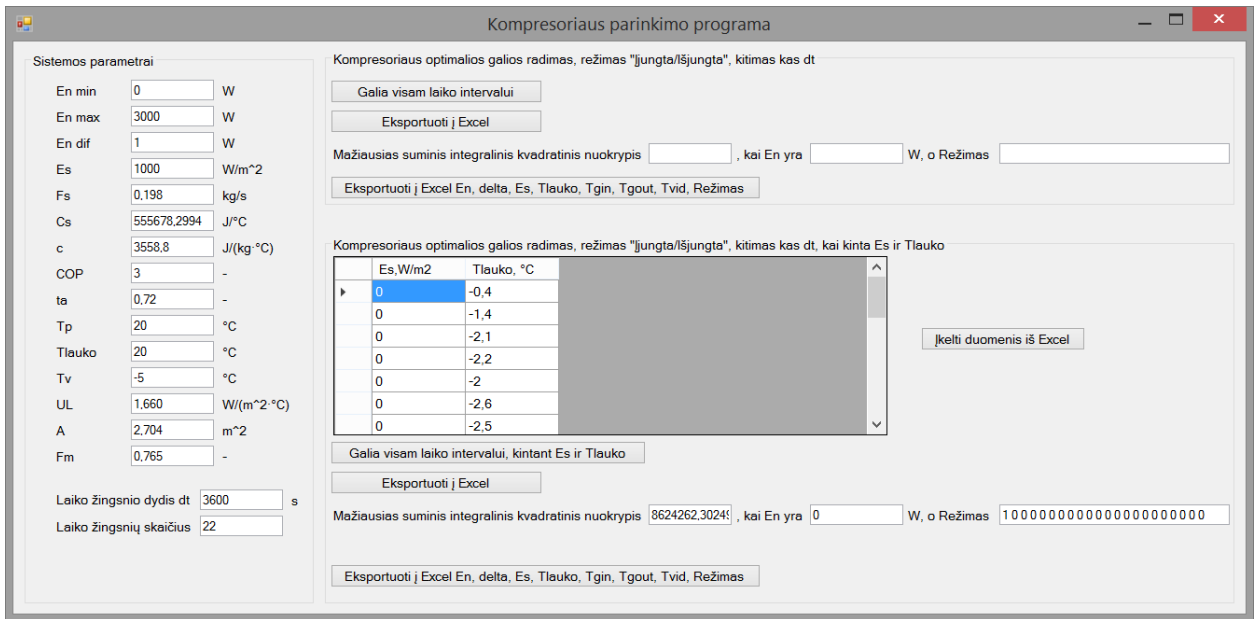
5.5. Kompresoriaus parinkimas, kai kinta saulės apšvieta *Es* ir lauko temperatūra *Tlauko*.

Sukurtoji programa turi funkciją – „Microsoft Excel“ duomenų įvedimas į skaičiuojamąją programą. Galima įvesti saulės apšvieta *Es* ir *Tlauko* temperatūros reikšmes kiekvienai valandai.

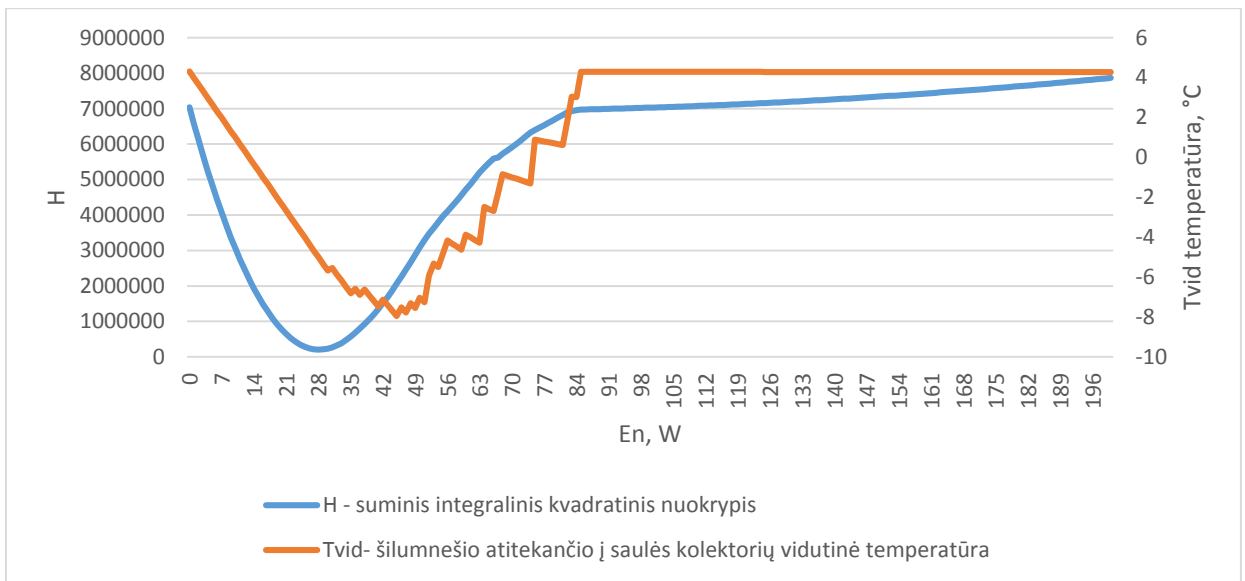
Pabandykime parinkti mūsų nagrinėjamai sistemai optimalios galios kompresorių, kai saulė šviečia mažiausiai – t.y. gruodžio, sausio vasario mėnesiais ir kai saulė šviečia daugiausiai - liepos mėnesį. Meteorologiniai duomenys pateikti 5.1 lentelė [21, 22]. Įkeliame į programą 5.1 lentelės duomenis. Programa kiekvienai iš nurodytų dienų parenka optimaliausią kompresoriaus galią ir valdymo režimą. Gautieji duomenis pateikiami sekančiuose paveiksluose.

5.1 lentelė. Meteorologiniai duomenys [21, 22]

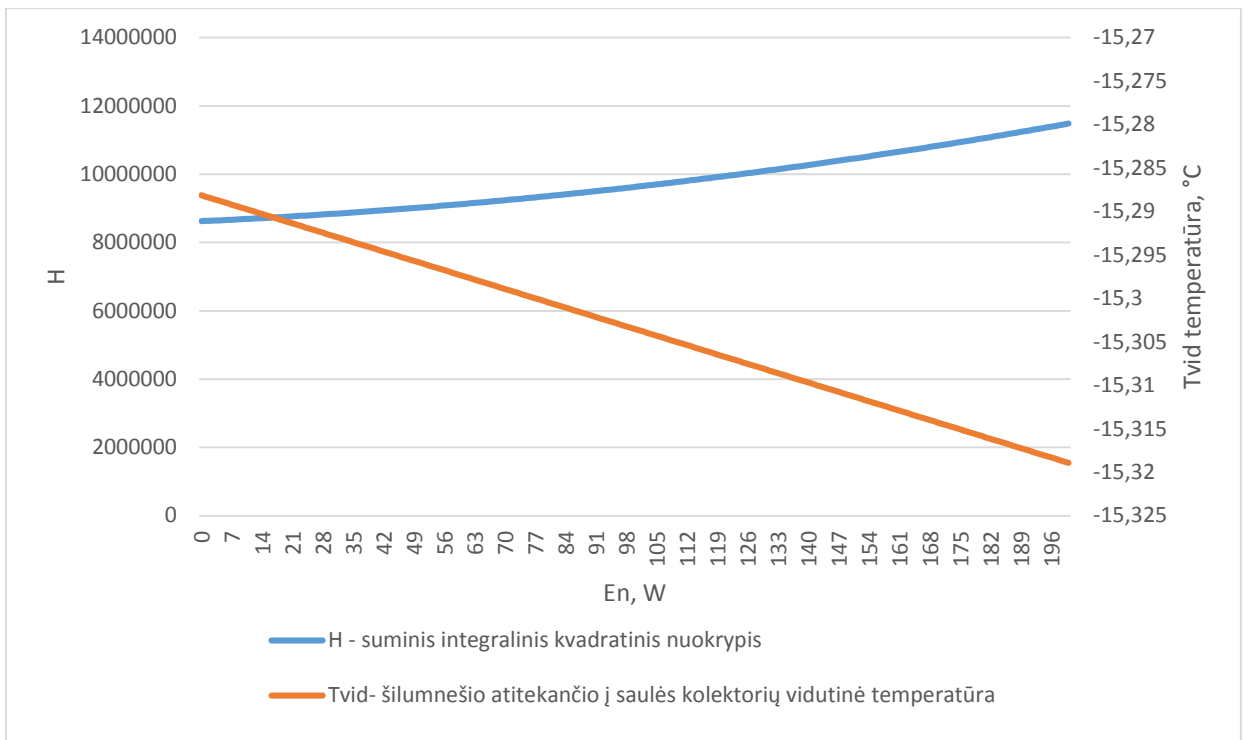
Stebėjimų laikas	2013-12-23		2014-01-23		2014-02-23		2012-07-27	
	Es, W/m ²	Tlauko, °C	Es, W/m ²	Tlauko, °C	Es, W/m ²	Tlauko, °C	Es, W/m ²	Tlauko, °C
01:00:00	0	6	0	-16,4	0	-0,4	0	16,6
02:00:00	0	6	0	-16,8	0	-1,4	0	16
03:00:00	0	6,1	0	-17,2	0	-2,1	0	15,4
04:00:00	0	5,5	0	-17,2	0	-2,2	0	17
05:00:00	0	5,3	0	-17,4	0	-2	10	19,1
06:00:00	0	5,8	0	-17,6	0	-2,6	34	22,3
08:00:00	0	5,6	0	-17,9	0	-2,5	100	23,5
09:00:00	3	5,5	0	-17,1	0	-2,3	380	26,5
10:00:00	13	5,7	34	-16	102	-0,3	550	27,1
11:00:00	19	5,3	133	-14,7	193	1,7	600	28,2
12:00:00	24	4,6	226	-14	323	4,4	800	29
13:00:00	28	4,6	275	-12,8	413	7	1198	28,5
14:00:00	38	4,7	281	-12,7	454	8,6	1101	28,9
15:00:00	51	4	244	-13,2	416	7,8	1005	28
16:00:00	7	3,9	68	-13,9	281	7,2	847	28
17:00:00	0	3,4	4	-13,9	176	5	630	27,3
18:00:00	0	2,8	0	-14,4	5	2,2	368	25
19:00:00	0	2,6	0	-14,5	0	1,3	189	24,6
20:00:00	0	2,3	0	-14,6	0	1,1	70	22,4
21:00:00	0	2,2	0	-15,1	0	0	10	19,8
22:00:00	0	1,5	0	-15,6	0	0	0	18
23:00:00	0	0,9	0	-16	0	0	0	17,8



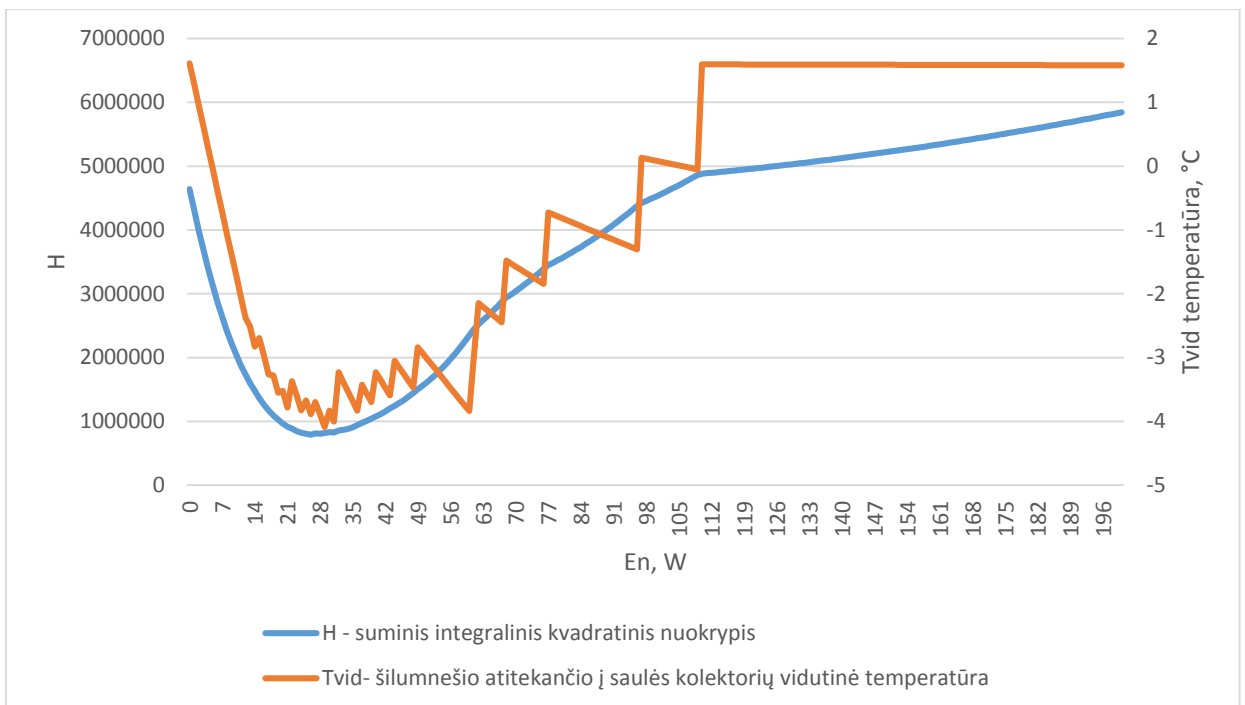
5.11 pav. Programos langas po įvestų duomenų ir atlikto skaičiavimo



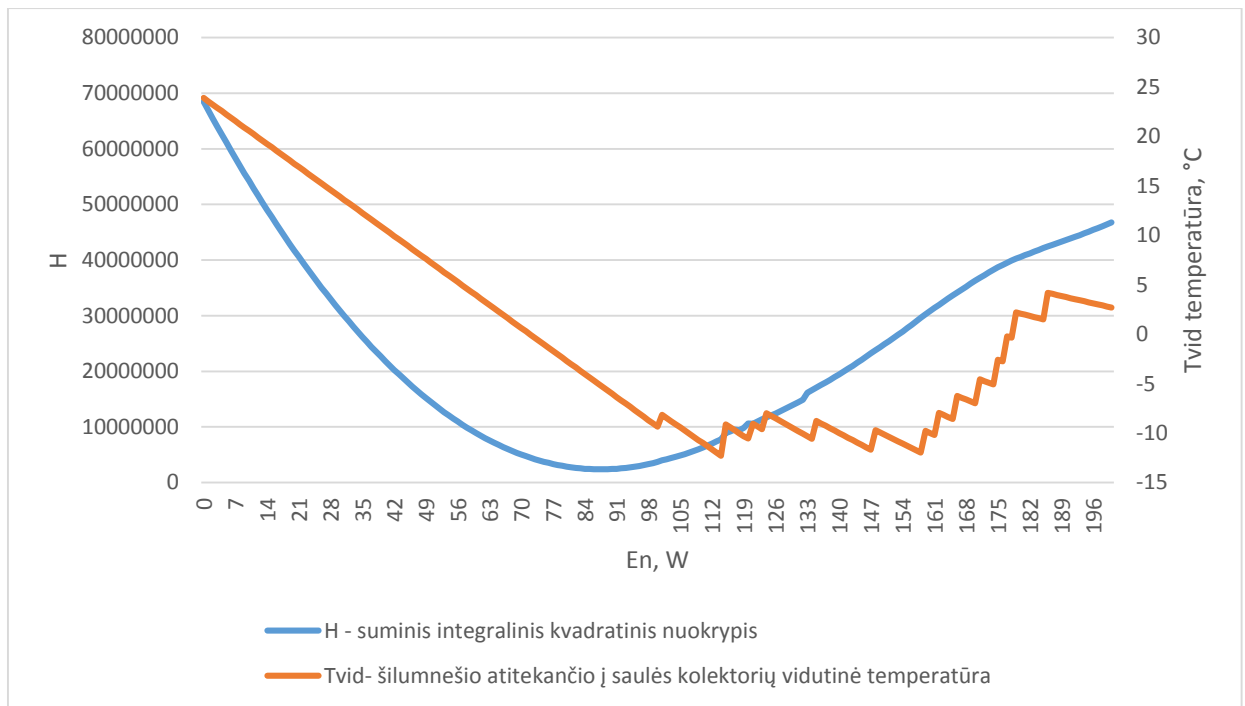
5.12 pav. T_{vid} ir H pokytis 2013-12-23 dieną. Optimalus $En = 28$ W, gaunamas darbinis režimas „1 1“, kai $T_v = -5$ °C



5.13 pav. *Tvid* ir *H* pokytis 2014-01-23 dieną. Optimalus $En = 0$ W, gaunamas darbinis režimas „visada išjungta“, kai $Tv = -5$ °C



5.14 pav. *Tvid* ir *H* pokytis 2014-02-23 dieną. Optimalus $En = 26$ W, gaunamas darbinis režimas „1 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1“, kai $Tv = -5$ °C



5.15 pav. T_{vid} ir H pokytis 2014-07-27 dieną. Optimalus $E_n = 88$ W, gaunamas darbinis režimas „1 1“, kai $T_v = -5$ °C

Prie tokių 5.1 lentelės saulės apšvietos ir lauko temperatūros verčių pasirinktosios sistemos kompresoriaus galia E_n kinta intervale 0 iki 88 W.

5.6. Kombinuotos saulės kolektorių ir šilumos siurblio sistemos palyginimas su sistema, kurioje yra tik saulės kolektorius

Pernešamos šilumos kiekis iš saulės kolektoriaus į akumuliacinę talpą arba į šilumos siurblio garintuvą priklauso nuo šilumnešio savitosios šilumos c [J/(kg·°C)], šilumnešio masės m [kg] ir temperatūrų skirtumo tarp saulės kolektoriaus išėjimo ir įėjimo temperatūrų $T_{cout} - T_{cin}$. Šiluminės energijos (šilumos) kiekis apskaičiuojamas pagal formulę:

$$Q = c \cdot m \cdot (T_{cout} - T_{cin}) \quad (41)$$

Kadangi tiek kombinuotoje saulės kolektorių ir šilumos siurblio sistemoje, tiek sistemoje, kurioje yra tik saulės kolektorius šilumnešis ir šilumnešio masė yra tokie pat, todėl pernešamos šilumos kiekis priklausys tik nuo temperatūrų skirtumo $T_{cout} - T_{cin}$.

Tarkime, kad visi sistemos parametrai tiek kombinuotai, tiek saulės kolektoriaus sistemai yra tokie patys, o skiriasi tik T_{cin} temperatūros.

Iš atliktų modeliavimų, žinome, kad naudojant optimalų šilumos siurblio valdymą, kombinuotoje sistemoje $T_{cin_kombinuota} = T_{vid} = -5\text{ }^{\circ}\text{C}$.

Iš patirties žinome, kad sistemoje, kurioje yra tik saulės kolektorius į saulės kolektorių grįžta, tokia temperatūra, kuri yra nusistovėjusi akumuliacinėje talpoje $T_{cin_tik_kolektorius} = T_b$. Paprastai vandens temperatūra, akumuliacinėje talpoje, prieš pradėdant kaitinti saulei būna lygi vandentiekio šalto vandens temperatūrai – $8\text{ }^{\circ}\text{C}$.

Tarkime kolektoriaus išėjimo temperatūra atitinka lauko temperatūrą $T_{lauko} = 20\text{ }^{\circ}\text{C}$. Tad pradiniu laiko momentu šilumos kiekių santykis tarp dviejų lyginamų sistemų bus:

$$Q_{kombinuota} / Q_{tik_kolektorius} = (T_{cout} - T_{cin_kombinuota}) / (T_{cout} - T_{cin_tik_kolektorius}) \quad (42)$$

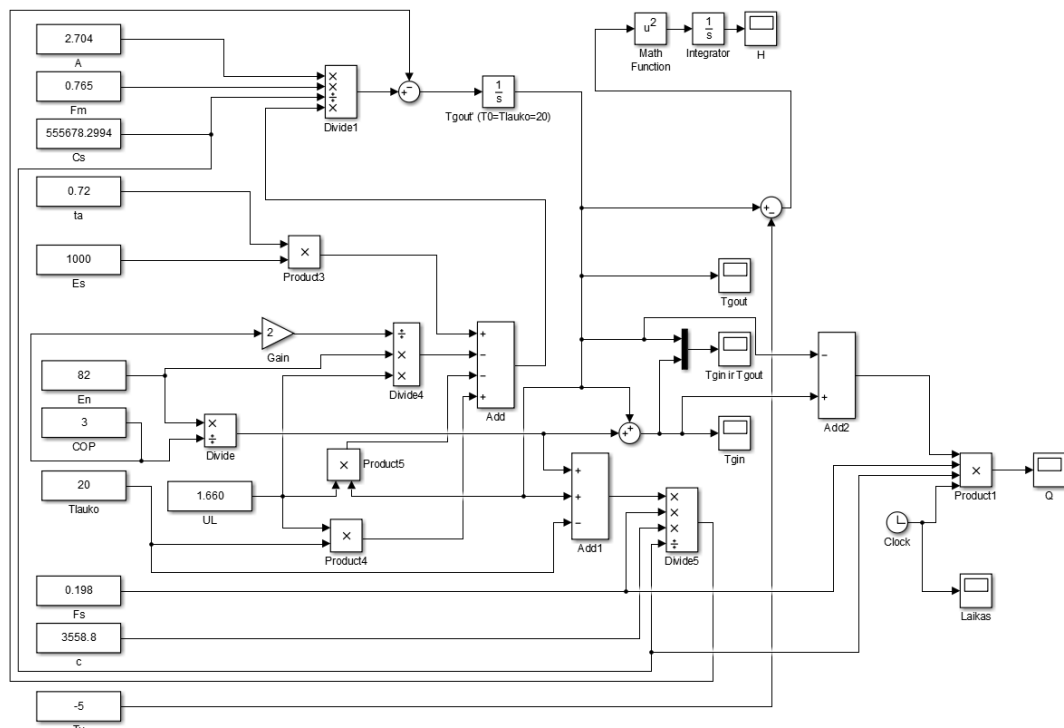
$$Q_{kombinuota} / Q_{tik_kolektorius} = (20 - (-5)) / (20 - 8) = 25/12 \approx 2,083$$

Kaip matyti iš gauto rezultato kombinuota saulės kolektoriaus ir šilumos siurblio sistema pagamina 2,083 kartus daugiau energijos nei sistema, kurioje yra tik kolektorius.

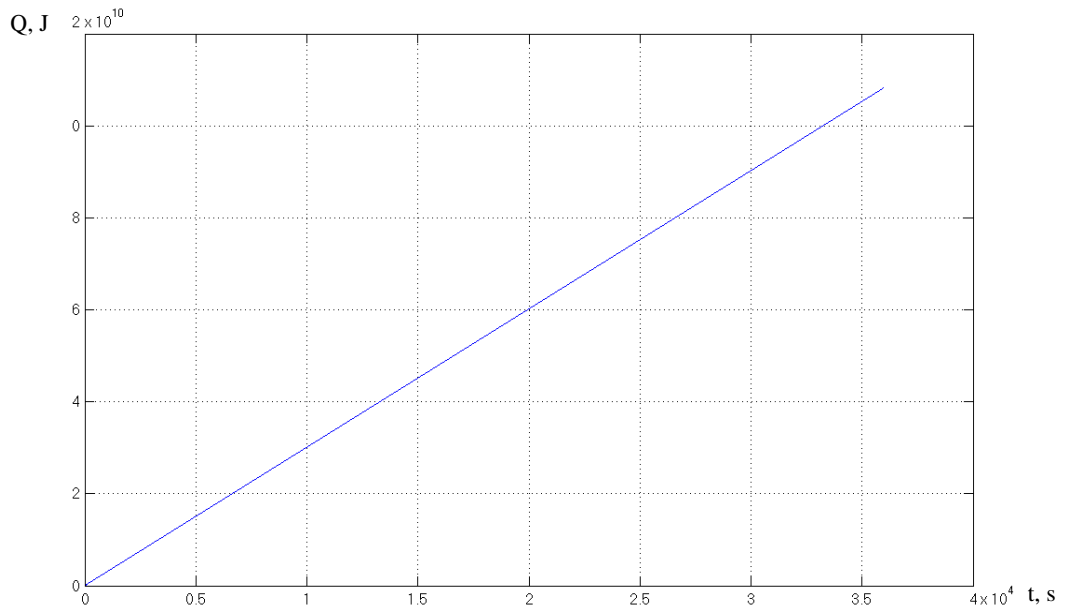
Kai sistemos pasiekia stacionarų režimą sistemoje, kurioje yra tik saulės kolektorius gauname tokią situaciją $T_{cout} = T_{cin_tik_kolektorius} = T_b$, o $T_{cout} - T_{cin_tik_kolektorius} = 0\text{ }^{\circ}\text{C}$, todėl šioje sistemoje saulės kolektorius daugiau nebegali pernešti šiluminės energijos į akumuliacinę talpą.

Kombinuotoje sistemoje, kur veikia šilumos siurblys to neatsitinka, nes nuolat palaikoma $T_{cin_kombinuota} = -5\text{ }^{\circ}\text{C}$, tad saulės kolektorius visada gali pernešti sugeneruojama šilumos kiekį.

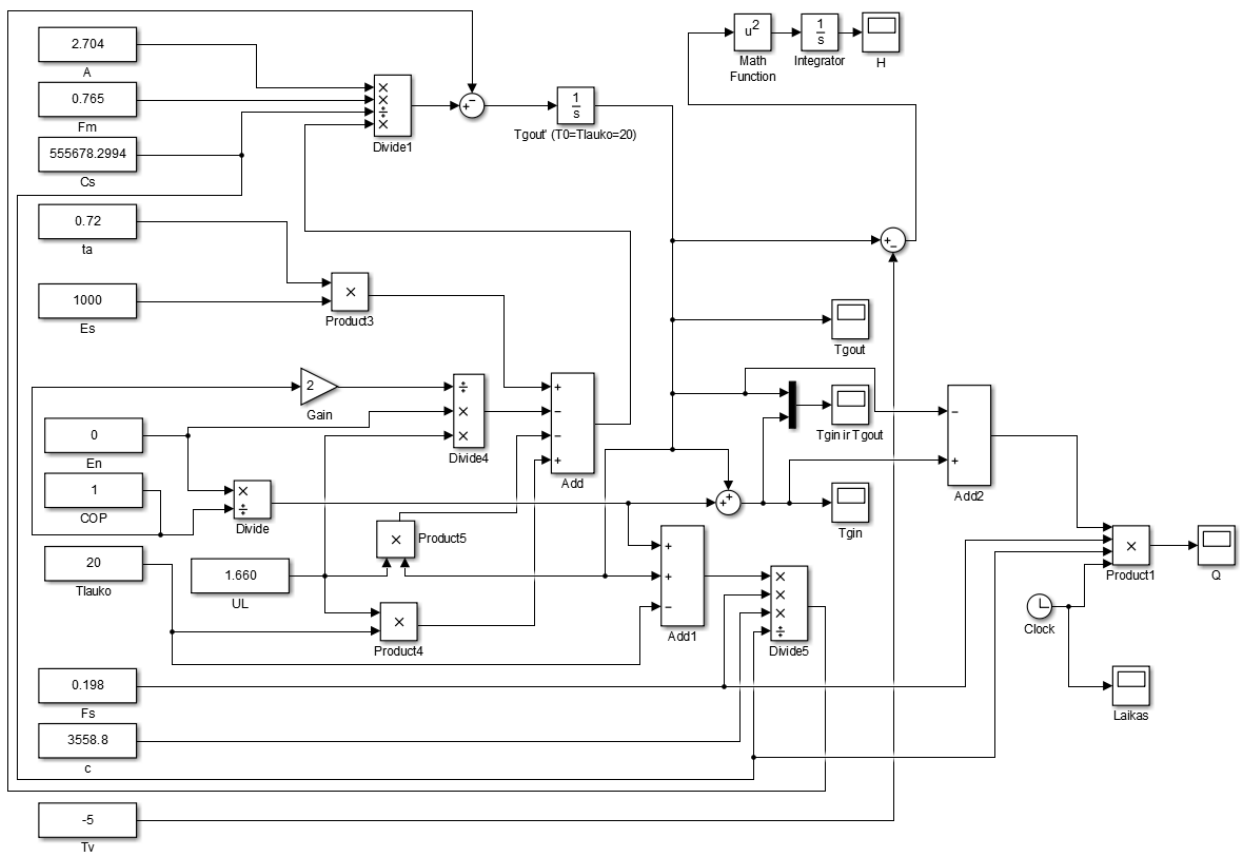
Tokius pačius rezultatus gauname ir modeliuojant šias dvi sistemas „Matlab Simulink“ pagalba. 5.16 pateiktas „Matlab Simulink“ modelis, kai veikia kombinuota sistema.



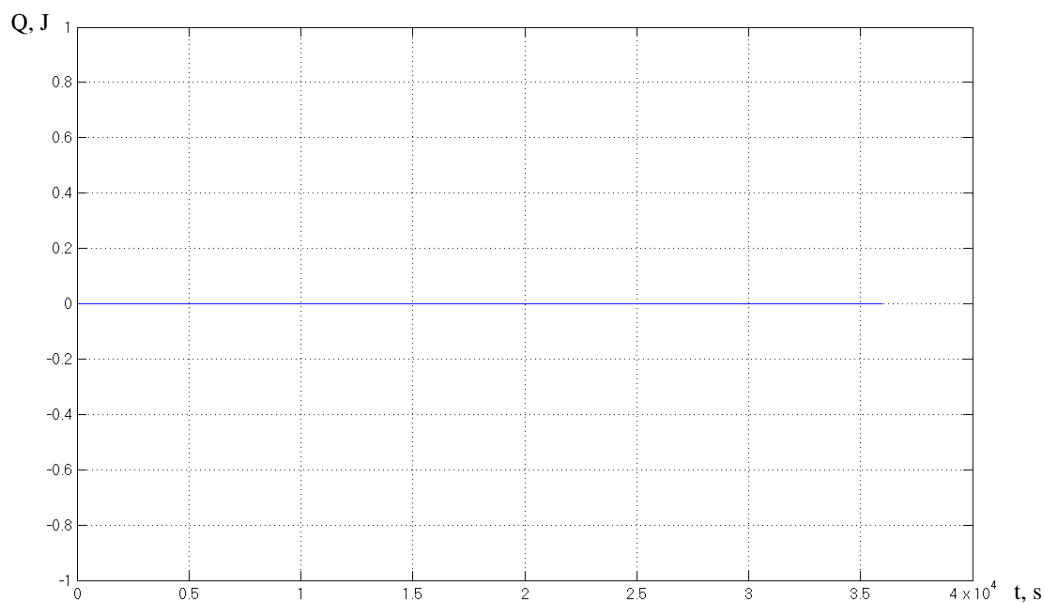
5.16 pav. Kombinuotos sistemos modelis



5.17 pav. Kombinuotos sistemos šilumos kiekio $Q_{kombinuota}$ grafikas per 10 h



5.18 pav. Sistemos, kurioje yra tik saulės kolektorius modelis



5.19 pav. Sistemos, kurioje yra tik saulės kolektorius
šilumos kiekio $Q_{tik_kolektorius}$ grafikas per 10 h

5.17 ir 5.19 grafikai paliudija, kad saulės kolektoriaus efektyvumas yra pagerinamas, kai saulės kolektoriaus šilumnešis yra papildomai vėsinamas šilumos siurblio pagalba, generuojamas pastovus temperatūrų skirtumas $T_{cout} - T_{cin_kombinuota} = 7,33 \text{ } ^\circ\text{C}$.

Iš 5.17 grafiko matyti, kad kai nenaudojamas šilumos siurblys, o sistemoje nėra nuostolių, tada $T_{cout} - T_{cin_tik_kolektorius} = 0 \text{ } ^\circ\text{C}$, o $Q_{tik_kolektorius} = 0 \text{ J}$.

IŠVADOS IR PASIŪLYMAI

1. Sudarytas modeliujamos sistemos matematinis ir „Matlab Simulink“ modelis. Pasirinktas sistemos optimizavimo kriterijus H – integralinis kvadratinis nuokrypis, kuris yra supaprastinta Pontriagino minimumo (maksimumo) principo Hamiltoniano forma. Taikant šį parametą temperatūros T_{vid} paklaida nuo užduotosios vertės T_v buvo mažesnė nei 0,5 %.
2. Suminio integralinio kvadratinio nuokrypio pagalba buvo parenkama optimali šilumos siurblio kompresoriaus galia ir nustatomas optimalus jo darbo režimas, kai sistemos modeliavimo laikas $\Delta t = 10$ h.
3. Sistemos valdymas atliekamas prognozuojant visos dienos (10 h) optimalaus šilumos siurblio kompresoriaus darbą režimą ir optimalią jo galią, įvertinant kiekvieno ankstesnio laiko intervalo sistemos parametrus. Dėl šios priežasties sistema dirba su vėlinimu.
4. Sukurtas ir patikrintas programos algoritmas C# kalboje su realių sistemų objektų parametrais, statistiniais bei archyviniais duomenimis. Nustatyta, kad esant žemai lauko temperatūrai ir nedidelei saulės apšvietai (žiemos periodas), šilumos siurblio kompresorius visą nagrinėjamą laiko tarpą (10 h) neįjungiamas - $E_n = 0$ W. Esant didelei apšvietai ir aukštai lauko temperatūrai (vasaros periodas), kompresorių reikia laikyti pastoviai įjungtą, o jo galia gaunama didžiausia $E_n = 88$ W.
5. Nustatyta, kad pagal pasirinktus kombinuotos saulės kolektoriaus ir šilumos siurblio sistemos parametrus: saulės apšvietai E_s – nekinta, šilumnešio nustatyta temperatūra yra diapazone $T_v = -20 \div 20$ °C, kompresoriaus parenkama optimali galia E_n kinta diapazone 0 – 128 W, tada bus pasiekiamas optimalus sistemos darbas.

LITERATŪRA

1. A. Genutis, A. Gulbinas, K. Navickas, I. Šateikis. „Atsinaujinantus energijos šaltiniai“. Akademija: 2008. 98 psl.
2. A. Genutis, K. Navickas, G. Rutkauskas, I. Šateikis. Atsinaujinančios ir alternatyviosios energijos naudojimas šilumos gamybai. Kaunas: Technologija, 2003. 112 psl.
3. P. Omojaro, C. Breitung. Direct expansion solar assisted heat pumps: A review of applications and recent research. Prieiga internete: <http://www.sciencedirect.com/science/article/pii/S1364032113000609> [žiūrėta 2015-03-14]
4. Flat Plate Solar Collector. Prieiga internete: <http://www.solarmastertech.com/page.php?cateid=232> [žiūrėta 2014-04-20]
5. A. Kavolynas. Plokščiojo saulės kolektoriaus tyrimas. Magistrantūros studijų baigiamasis darbas. Akademija: 2005. 102 psl.
6. I. Šateikis, A. Genutis. Saulės kolektoriai. Raudondvaris: Milga, 1999. 38 psl.
7. G. Rutkauskas. Automatizuoto saulės kolektoriaus vandeniui šildyti tyrimo rezultatai. Pranešimų santrauka. Lietuvos žemės ūkio mechanizacijos institutas. Raudondvaris: Milga, 1995. 26 psl.
8. Heat pipe solar collector scheme. Prieiga internete: <http://www.prosunindia.com/images/hwheat1.gif> [žiūrėta 2015-01-15]
9. U- pipe saulės kolektoriaus schema. Prieiga internete: http://www.saulėsbroliai.lt/wp-content/uploads/2014/10/1_1zjb360.jpg [žiūrėta 2015-01-15]
10. I. Šateikis. Tūrinio saulės kolektoriaus vandeniui šildyti tyrimas. LŽŪI Instituto ir LŽŪU mokslo darbai LŽŪU. 1998. 30 (1), P.131-144.
11. B. Kavolėlis. Savos gamybos saulės kolektoriai vandeniui šildyti. VI „Energetikos agentūra“. Raudondvaris: Milga, 1996. 35 psl.
12. Šilumos siurblio veikimo principas. Prieiga internete: <http://www.kond.lt/informacija/silumos-siurblio-veikimo-principas.html> [žiūrėta 2014-03-18].
13. M. Jakubčionis. Šiluminės technikos pagrindai. Mokomoji knyga. Vilnius: Vilniaus pedagoginio universiteto leidykla, 2011. 208 psl.
14. Optimalus ir adaptyvus elektromechaninių sistemų valdymas (T125M162). Paskaitų medžiaga. 2013.
15. A. Giffin, R. Urniežius. Simultaneous State and Parameter Estimation Using Maximum Relative Entropy with Nonhomogenous Differential Equation Constraints. Published: 17

- September 2014. Prieiga internete: <http://www.mdpi.com/1099-4300/16/9/4974> [žiūrėta 2015-05-10].
16. A. Giffin, R. Urniežius. The Kalman Filter Revisited Using Maximum Relative Entropy. 19 February 2014. Prieiga internete: <http://www.mdpi.com/1099-4300/16/2/1047> [žiūrėta 2015-05-11].
 17. Rectangle approximation method. Prieiga internete: http://en.wikipedia.org/wiki/Rectangle_method [žiūrėta 2015-05-14]
 18. Vakuuminio saulės kolektoriaus QBJ-90/3.90/0 charakteristikos. Prieiga internete: <http://el.sprendimas.net/saules-energijos-panaudojimas/vakuuminis-saules-kolektorius> [žiūrėta 2014-02-02]
 19. Heat Transfer Fluids for Solar Water Heating Systems. Prieiga internete: <http://energy.gov/energysaver/articles/heat-transfer-fluids-solar-water-heating-systems> [žiūrėta 2014-04-30]
 20. Propylene Glycol based Heat-Transfer Fluids. Prieiga internete: http://www.engineeringtoolbox.com/propylene-glycol-d_363.html [žiūrėta 2014-04-30]
 21. R. Endrikis. Paduodamos šilumos gyvenamajam namui valdymo tyrimas įvertinant saulės ir vėjo duomenis. Magistrinis darbas. 2015.
 22. G. Meška. Saulės kolektorių sistemos Šiaulių mieste energetinio efektyvumo tyrimas. Šiauliai, 2013. 70 psl.

PRIEDAI

1 PRIEDAS. WOLFRAM MATHEMATICA 10“ PROGRAMOS ALGORITMAS MATEMATINIŲ IŠRAIŠKŲ RADIMUI

```

eq1=En-COP *(Tgin[t]-Tgout[t])

Print["Tgin lygtis"]
solTgin=FullSimplify[Solve[eq1==0,Tgin[t]]][[1]]

Print["Tgin išvestines lygtis"]
solTginisv=FullSimplify[D[solTgin,t]]

eq2=1/Cs A*Fm*(Es*ta - UL* ((Tgin[t]+Tgout[t])/2-Tlauko))-(Fs
*c)/Cs (Tgin[t]-Tlauko)-Tgout'[t]
eq22=FullSimplify[eq2/.solTgin/.solTginisv]

Print["Tgout lygtis"]
soldT=FullSimplify[DSolve[{eq22==0,Tgout[0]==Tp},Tgout[t],t]][[1]]

Print["Kriterijus"]
eqHH=((Tgout[t]-Tv)/.soldT)^2

Print["vidutinis kvadratinis nuokrypis"]
eqInt=FullSimplify[ $\int_0^{\Delta t} eqHH dt$ ]

Print["Saules naudinga galia"]
SaulesGalia = A*Fm*(Es*ta - UL* ((Tgin[t]+Tgout[t])/2-Tlauko))

vals={En -> 50,Es->1000,Fs->0.198,Cs->444648.7,c ->3558.8,COP-
>3,Tp->20,Tlauko->20,Tv->-5, t->1800, A->2.704, Fm->0.765, ta-
>0.72,UL->1.660}

vals2={En -> 50,Es->1000,Fs->0.198,Cs->444648.7,c ->3558.8,COP-
>3,Tp->20,Tlauko->20,Tv->-5,  $\Delta t$ ->1800, A->2.704, Fm->0.765, ta-
>0.72,UL->1.660}

Print["Tgout lygtis su skaiciais"]
soldTskaičius=N[FullSimplify[soldT/.vals]]

Print["Tgin lygtis su skaiciais"]
solTginskaičius=N[FullSimplify[solTgin/.vals]]

Print["vidutinis kvadratinis nuokrypis su skaiciais"]
eqIntskaičius=N[eqInt/.vals2]

Print["Saules naudinga galia su skaiciais"]
SaulesGaliaSkaicius=N[FullSimplify[SaulesGalia/.vals]]

```

En - COP (Tgin[t] - Tgout[t])

Tgin lygtis

$$\left\{ Tgin[t] \rightarrow \frac{En}{COP} + Tgout[t] \right\}$$

{Tgin'[t] → Tgout'[t]}

$$-\frac{c Fs (-Tlauko + Tgin[t])}{Cs} + \frac{A Fm (Es ta - UL (-Tlauko + \frac{1}{2} (Tgin[t] + Tgout[t])))}{Cs} - Tgout'[t]$$

$$-\frac{c Fs (\frac{En}{COP} - Tlauko + Tgout[t])}{Cs} + \frac{A Fm (Es ta - \frac{En UL}{2 COP} + Tlauko UL - UL Tgout[t])}{Cs} - Tgout'[t]$$

Tgout lygtis

$$\left\{ Tgout[t] \rightarrow \frac{1}{2 COP (c Fs + A Fm UL)} \right.$$

$$e^{-\frac{t (c Fs + A Fm UL)}{Cs}} \left(2 c Fs (En - COP Tlauko + e^{\frac{t (c Fs + A Fm UL)}{Cs}} (-En + COP Tlauko) + COP Tp) + \right.$$

$$\left. \left. A Fm (En UL - 2 COP (Es ta + (Tlauko - Tp) UL) + e^{\frac{t (c Fs + A Fm UL)}{Cs}} (-En UL + 2 COP (Es ta + Tlauko UL)) \right) \right\}$$

Kriterijus

$$\left(-Tv + \frac{1}{2 COP (c Fs + A Fm UL)} \right.$$

$$e^{-\frac{t (c Fs + A Fm UL)}{Cs}} \left(2 c Fs (En - COP Tlauko + e^{\frac{t (c Fs + A Fm UL)}{Cs}} (-En + COP Tlauko) + COP Tp) + A Fm \right.$$

$$\left. \left. \left(En UL - 2 COP (Es ta + (Tlauko - Tp) UL) + e^{\frac{t (c Fs + A Fm UL)}{Cs}} (-En UL + 2 COP (Es ta + Tlauko UL)) \right) \right) \right)^2$$

vidutinis kvadratinis nuokrypis

$$\frac{1}{8 COP^2 (c Fs + A Fm UL)^3}$$

$$\left(Cs (2 c Fs (En + COP (-Tlauko + Tp)) + A Fm (En UL - 2 COP (Es ta + Tlauko UL - Tp UL)))^2 - \right. \\ Cs e^{-\frac{2 (c Fs + A Fm UL) \Delta t}{Cs}} (2 c Fs (En + COP (-Tlauko + Tp)) + A Fm (En UL - 2 COP (Es ta + Tlauko UL - Tp UL)))^2 - \\ 4 Cs (2 c Fs (En + COP (-Tlauko + Tp)) + A Fm (En UL - 2 COP (Es ta + Tlauko UL - Tp UL))) \\ (2 c Fs (En + COP (-Tlauko + Tv)) + A Fm (En UL - 2 COP (Es ta + Tlauko UL - Tv UL))) + \\ 4 Cs e^{-\frac{(c Fs + A Fm UL) \Delta t}{Cs}} (2 c Fs (En + COP (-Tlauko + Tp)) + A Fm (En UL - 2 COP (Es ta + Tlauko UL - Tp UL))) \\ (2 c Fs (En + COP (-Tlauko + Tv)) + A Fm (En UL - 2 COP (Es ta + Tlauko UL - Tv UL))) + 2 (c Fs + A Fm UL) \\ \left. (2 c Fs (En + COP (-Tlauko + Tv)) + A Fm (En UL - 2 COP (Es ta + Tlauko UL - Tv UL)))^2 \Delta t \right)$$

Saules naudinga galia

$$A Fm (Es ta - UL (-Tlauko + \frac{1}{2} (Tgin[t] + Tgout[t])))$$

{En → 50, Es → 1000, Fs → 0.198, Cs → 444 649., c → 3558.8, COP → 3, Tp → 20,
Tlauko → 20, Tv → -5, t → 1800, A → 2.704, Fm → 0.765, ta → 0.72, UL → 1.66}

{En → 50, Es → 1000, Fs → 0.198, Cs → 444 649., c → 3558.8, COP → 3, Tp → 20,
Tlauko → 20, Tv → -5, Δt → 1800, A → 2.704, Fm → 0.765, ta → 0.72, UL → 1.66}

Tgout lygtis su skaiciais

$$\{Tgout[1800.] \rightarrow 6.30355\}$$

Tgin lygtis su skaiciais

$$\{Tgin[1800.] \rightarrow 16.6667 + Tgout[1800.]\}$$

vidutinis kvadratinis nuokrypis su skaiciais

443822.

Saules naudinga galia su skaiciais

$$1558.04 - 1.7169 Tgin[1800.] - 1.7169 Tgout[1800.]$$

2 PRIEDAS. PROGRAMOS KODAS PARAŠYTAS C# KALBA

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using ExportToExcel;
using System.Data.OleDb;

namespace Kompresoriaus_parinkimo_programa
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        int Enmin;
        int Enmax;
        double Es;
        int N;
        double Fs;
        double Cs;
        double c;
        double COP;
        double ta;
        double UL;
        double Fm;
        double A;
        double Tp;
        double Tlauko;
        double Tgin;
        double Tgout;
        double delta;
        double deltaDabartine;
        double TvidDabartine;
        double Tv;
        double En;
        double EnP;
        int dt;
        int En_diff;

        List<EnConstClass> listEnConst = new List<EnConstClass>();

        public class EnConstClass
        {
            public string Galia { get; set; }
            public string Suminis_Nuokrypis { get; set; }
            public string Vid_Temperatura { get; set; }
        }

        List<EnConstOnOffClass> listEnConstOnOff = new List<EnConstOnOffClass>();
        List<EnConstOnOffClass> listEnConstOnOffEs = new List<EnConstOnOffClass>();

        private class EnConstOnOffClass
        {
            public string En_galia { get; set; }
            public string delta_suminis_integralinis_kvadratinis_nuokrypis { get; set; }
            public string Tvid_Temperatura { get; set; }
            public string Režimas { get; set; }
        }

        List<listEnConstOnOffDabartineClass> listEnConstOnOffDabartine = new List<listEnConstOnOffDabartineClass>();
        List<listEnConstOnOffDabartineClass> listEnConstOnOffDabartineEs = new List<listEnConstOnOffDabartineClass>();

        private class listEnConstOnOffDabartineClass
        {
            public string Laikas { get; set; }
            public string En_galia { get; set; }
            public string delta_integralinis_kvadratinis_nuokrypis { get; set; }
            public string Es_saulės_apšvieta { get; set; }
            public string Tlauko_temperatūra { get; set; }
            public string Tgin_temperatūra { get; set; }
            public string Tgout_temperatūra { get; set; }
            public string Tvid_temperatūra { get; set; }
            public string Režimas { get; set; }
        }

        List<All> listAll = new List<All>();
        public class All
        {
            public string Laikas { get; set; }
            public string Galia { get; set; }
            public string Nuokrypis { get; set; }
            public string Pradine_Temperatura { get; set; }
            public string Vid_Temperatura { get; set; }
        }

        public void textBoxEnmin_TextChanged(object sender, EventArgs e)
        {
            int Enmin;
            if (!int.TryParse(textBoxEnmin.Text, out Enmin))
            {
                // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
                MessageBox.Show("Prašau įveskite skaičių!");
            }
        }

        public void textBoxEnmax_TextChanged(object sender, EventArgs e)
        {
            int Enmax;
            if (!int.TryParse(textBoxEnmax.Text, out Enmax))
            {
                // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
                MessageBox.Show("Prašau įveskite skaičių!");
            }
        }
    }
}
```

```

public void textBoxEs_TextChanged(object sender, EventArgs e)
{
    double Es;
    if (!double.TryParse(textBoxEs.Text, out Es))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxN_TextChanged(object sender, EventArgs e)
{
    int N;
    if (!int.TryParse(textBoxN.Text, out N))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxFs_TextChanged(object sender, EventArgs e)
{
    double Fs;
    if (!double.TryParse(textBoxFs.Text, out Fs))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxCs_TextChanged(object sender, EventArgs e)
{
    double Cs;
    if (!double.TryParse(textBoxCs.Text, out Cs))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxc_TextChanged(object sender, EventArgs e)
{
    double c;
    if (!double.TryParse(textBoxc.Text, out c))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxk_TextChanged(object sender, EventArgs e)
{
    double COP;
    if (!double.TryParse(textBoxCOP.Text, out COP))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxta_TextChanged(object sender, EventArgs e)
{
    double ta;
    if (!double.TryParse(textBoxta.Text, out ta))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxUL_TextChanged(object sender, EventArgs e)
{
    double UL;
    if (!double.TryParse(textBoxUL.Text, out UL))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxA_TextChanged(object sender, EventArgs e)
{
    double A;
    if (!double.TryParse(textBoxA.Text, out A))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxFm_TextChanged(object sender, EventArgs e)
{
    double Fm;
    if (!double.TryParse(textBoxFm.Text, out Fm))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxTp_TextChanged(object sender, EventArgs e)
{
    double Tp;
    if (!double.TryParse(textBoxTp.Text, out Tp))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public void textBoxTlauko_TextChanged(object sender, EventArgs e)
{
    double Tlauko;
    if (!double.TryParse(textBoxTlauko.Text, out Tlauko))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

private void textBoxTv_TextChanged(object sender, EventArgs e)
{
    double Tv;
    if (!double.TryParse(textBoxTv.Text, out Tv))
}

```

```

    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

private void textBoxdt_TextChanged(object sender, EventArgs e)
{
    int dt;
    if (!int.TryParse(textBoxdt.Text, out dt))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

private void textBoxEn_diff_TextChanged(object sender, EventArgs e)
{
    int En_diff;
    if (!int.TryParse(textBoxEn_diff.Text, out En_diff))
    {
        // Jei teksto lauke įvestas ne skaičius, pranešti vartotojui
        MessageBox.Show("Prašau įveskite skaičių!");
    }
}

public double Calc_Hamiltonian(int En_loc, out double Tvid)
{
    //Tp = Tgin = Tgout = double.Parse(textBoxTp.Text);
    Tp = Tgin = Tgout;
    delta = 0;
    Tvid = 0;

    deltaDabartine = 0;
    TvidDabartine = 0;

    StringBuilder builder = new StringBuilder();
    foreach (int EnValue in m_EnOptimalus)
    {
        builder.Append(EnValue).Append(" ");
    }
    string rezimas = builder.ToString();

    for (int t = 0; t < N; t++)
    {
        En = m_EnOptimalus[t] * (double)En_loc;
        Tgout = (2 * c * Fs * (En - COP * Tlauko + Math.Exp((dt * (c * Fs + A * Fm * UL)) / Cs) * (-En + COP * Tlauko) + COP * Tp) + A * Fm * (En * UL - 2 * COP * (Es * ta + (Tlauko - Tp) * UL) + Math.Exp((dt * (c * Fs + A * Fm * UL)) / Cs) * (-En * UL + 2 * COP * (Es * ta + Tlauko * UL)))) / (2 * COP * Math.Exp((dt * (c * Fs + A * Fm * UL)) / Cs) * (c * Fs + A * Fm * UL));

        Tgin = En / COP + Tgout;
        delta += (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL)), 2) - (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))), 2)) / Math.Exp((2 * (c * Fs + A * Fm * UL) * dt) / Cs) - 4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL))) + (4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL)))) / Math.Exp(((c * Fs + A * Fm * UL) * dt) / Cs) + 2 * (c * Fs + A * Fm * UL) * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL))), 2) * dt) / (8 * Math.Pow(COP, 2) * Math.Pow(c * Fs + A * Fm * UL, 3));

        deltaDabartine = (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL)), 2) - (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))), 2)) / Math.Exp((2 * (c * Fs + A * Fm * UL) * dt) / Cs) - 4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL))) + (4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL)))) / Math.Exp(((c * Fs + A * Fm * UL) * dt) / Cs) + 2 * (c * Fs + A * Fm * UL) * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL))), 2) * dt) / (8 * Math.Pow(COP, 2) * Math.Pow(c * Fs + A * Fm * UL, 3));

        Tp = Tgout;
        Tvid += Tgout;

        TvidDabartine = Tvid / (t + 1);

        listEnConstOnOffDabartine.Add(new listEnConstOnOffDabartineClass { Laikas = (t * dt).ToString(), En_galia = En.ToString(), Režimas = rezimas, delta_integralinis_kvadratinis_nuokrypis = deltaDabartine.ToString(), Es_saulės_apsvieta = Es.ToString(), Tlauko_temperatūra = Tlauko.ToString(), Tgin_temperatūra = Tgin.ToString(), Tgout_temperatūra = Tgout.ToString(), Tvid_temperatūra = TvidDabartine.ToString() });
    }
    Tvid /= N;

    return delta;
}

private void buttonEnConstOnOff_Click(object sender, EventArgs e)
{
    listEnConstOnOff.Clear();
    listEnConstOnOffDabartine.Clear();

    dt = int.Parse(textBoxdt.Text);
    En_diff = int.Parse(textBoxEn_diff.Text);
    Enmin = int.Parse(textBoxEnmin.Text);
    Enmax = int.Parse(textBoxEnmax.Text);

    Es = double.Parse(textBoxEs.Text);
    N = int.Parse(textBoxN.Text);

    Fs = double.Parse(textBoxFs.Text);
    Cs = double.Parse(textBoxCs.Text);

    c = double.Parse(textBoxc.Text);

    COP = double.Parse(textBoxCOP.Text);

    ta = double.Parse(textBoxta.Text);
    UL = double.Parse(textBoxUL.Text);
    A = double.Parse(textBoxA.Text);
    Fm = double.Parse(textBoxFm.Text);

    Tp = Tgin = Tgout = double.Parse(textBoxTp.Text);
    Tlauko = double.Parse(textBoxTlauko.Text);
    Tv = double.Parse(textBoxTv.Text);

    m_EnOptimalus.Clear();
    int t;
}

```



```

for (t = 0; t < N; t++)
{
    m_EnOptimalus.Add(0);
}

int En_loc = Emin;
while (En_loc <= Enmax)
{
    for (t = 0; t < N; t++)
    {
        m_EnOptimalus[t] = 0;
    }

    double Tvid = 0;
    Calc_Hamiltonian(En_loc, out Tvid);

    double old_hamiltonian = delta;
    int candidate_j = -1;
    short candidate_verte = 0;
    int iteracijos = 0;
    bool done = false;
    while(!done)
    {
        iteracijos++;
        done = true;
        for(t = 0; t < N; t++)
        {
            short prev_value = m_EnOptimalus[t];
            if (prev_value == 0)
                m_EnOptimalus[t] = 1;
            else
                m_EnOptimalus[t] = 0;

            Calc_Hamiltonian(En_loc, out Tvid);

            if(delta < old_hamiltonian)
            {
                old_hamiltonian = delta;
                candidate_j = t;
                candidate_verte = m_EnOptimalus[t];
                done = false;
            }
            m_EnOptimalus[t] = prev_value;
        }
        if (!done)
            m_EnOptimalus[candidate_j] = candidate_verte;
    }

    Calc_Hamiltonian(En_loc, out Tvid);

    StringBuilder builder = new StringBuilder();
    foreach (int EnValue in m_EnOptimalus)
    {
        builder.Append(EnValue).Append(" ");
    }
    string rezimas = builder.ToString();

    listEnConstOnOff.Add(new EnConstOnOffClass { En_galia = En_loc.ToString(), delta_suminis_integralinis_kvadratinis_nuokrypis = delta.ToString(),
    Tvid_Temperaturą = Tvid.ToString(), Rezimas = rezimas });

    En_loc += En_diff;
}

double? minDelta = null;
double thisEn = 0;
string thisRezimas = "Neteisingi ivesties duomenys";
for (int i = 0; i < listEnConstOnOff.Count; i++)
{
    double thisDelta = Convert.ToDouble(listEnConstOnOff[i].delta_suminis_integralinis_kvadratinis_nuokrypis);
    if (!minDelta.HasValue || thisDelta < minDelta.Value)
    {
        minDelta = thisDelta;
        thisEn = Convert.ToDouble(listEnConstOnOff[i].En_galia);
        thisRezimas = listEnConstOnOff[i].Rezimas;
    }
}

textBoxMinDelta.Text = minDelta.ToString();
textBoxEnOnOff.Text = thisEn.ToString();
textBoxRezimas.Text = thisRezimas;
}

public List<short> m_EnOptimalus = new List<short>();

private void buttonExcelEnConstOnOff_Click(object sender, EventArgs e)
{
    CreateExcelFile.CreateExcelDocument(listEnConstOnOff, "Galia nepastovi ON OFF.xlsx");
}

private void buttonExcelEnConstOnOffDabartine_Click(object sender, EventArgs e)
{
    CreateExcelFile.CreateExcelDocument(listEnConstOnOffDabartine, "Galia nepastovi ON OFF kas dt.xlsx");
}

List<double> EsList = new List<double>();
List<double> TlaukoList = new List<double>();

private void buttonIkelimas_Click(object sender, EventArgs e)
{
    EsList.Clear();
    TlaukoList.Clear();

    OpenFileDialog openFileDialog1 = new OpenFileDialog();
    if (openFileDialog1.ShowDialog() == System.Windows.Forms.DialogResult.OK)
    {
        string pathConn = "Provider=Microsoft.ACE.OLEDB.12.0; Data Source=" + openFileDialog1.FileName + ";Extended Properties='Excel 12.0 Xml;HDR=Yes;';";
        OleDbConnection con = new OleDbConnection(pathConn);

        OleDbDataAdapter dataAdapter = new OleDbDataAdapter("Select * from [Sheet1$]", con);
        DataTable data = new DataTable();

        dataAdapter.Fill(data);
        dataGridViewEs.DataSource = data;
    }
}

```

```

foreach (DataRow row in data.Rows)
{
    EsList.Add(row.Field<double>(0));
    TlaukoList.Add(row.Field<double>(1));
}
}

public double Calc_Hamiltonian_Es(int En_loc, out double Tvid)
{
    Tp = Tgin = Tgout;
    delta = 0;
    Tvid = 0;

    deltaDabartine = 0;
    TvidDabartine = 0;

    StringBuilder builder = new StringBuilder();
    foreach (int EnValue in m_EnOptimalus)
    {
        builder.Append(EnValue).Append(" ");
    }
    string rezimas = builder.ToString();

    for (int t = 0; t < N; t++)
    {
        Es = EsList[t];
        Tlauko = TlaukoList[t];
        En = m_EnOptimalus[t];
        Tgout = (2 * c * Fs * (En - COP * Tlauko + Math.Exp((dt * (c * Fs + A * Fm * UL)) / Cs) * (-En + COP * Tlauko) + COP * Tp) + A * Fm * (En * UL - 2 * COP * (Es * ta + (Tlauko - Tp) * UL) + Math.Exp((dt * (c * Fs + A * Fm * UL)) / Cs) * (-En * UL + 2 * COP * (Es * ta + Tlauko * UL)))) / (2 * COP * Math.Exp(dt * (c * Fs + A * Fm * UL) / Cs) * (c * Fs + A * Fm * UL));
        Tgin = En / COP + Tgout;
        delta += (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL)), 2) - (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL)), 2)) / Math.Exp((2 * (c * Fs + A * Fm * UL) * dt) / Cs) - 4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL))) + (4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL)))) / Math.Exp(((c * Fs + A * Fm * UL) * dt) / Cs) + 2 * (c * Fs + A * Fm * UL) * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL)), 2) * dt) / (8 * Math.Pow(COP, 2) * Math.Pow(c * Fs + A * Fm * UL, 3));
        deltaDabartine = (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL)), 2) - (Cs * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL)), 2)) / Math.Exp((2 * (c * Fs + A * Fm * UL) * dt) / Cs) - 4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL))) + (4 * Cs * (2 * c * Fs * (En + COP * (-Tlauko + Tp)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tp * UL))) * (2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL)))) / Math.Exp(((c * Fs + A * Fm * UL) * dt) / Cs) + 2 * (c * Fs + A * Fm * UL) * Math.Pow(2 * c * Fs * (En + COP * (-Tlauko + Tv)) + A * Fm * (En * UL - 2 * COP * (Es * ta + Tlauko * UL - Tv * UL)), 2) * dt) / (8 * Math.Pow(COP, 2) * Math.Pow(c * Fs + A * Fm * UL, 3));
        Tp = Tgout;
        Tvid += Tgout;

        TvidDabartine = Tvid / (t + 1);

        listEnConstOnOffDabartineEs.Add(new listEnConstOnOffDabartineClass { Laikas = (t * dt).ToString(), En_galia = En.ToString(), Rezimas = rezimas, delta_integralinis_kvadratinis_nuokrypis = deltaDabartine.ToString(), Es_saulės_apšvieta = Es.ToString(), Tlauko_temperatūra = Tlauko.ToString(), Tgin_temperatūra = Tgin.ToString(), Tgout_temperatūra = Tgout.ToString(), Tvid_temperatūra = TvidDabartine.ToString() });
    }
    Tvid /= N;

    return delta;
}

private void buttonEsKintanti_Click(object sender, EventArgs e)
{
    listEnConstOnOffEs.Clear();
    listEnConstOnOffDabartineEs.Clear();

    dt = int.Parse(textBoxdt.Text);
    En_diff = int.Parse(textBoxEn_diff.Text);
    Enmin = int.Parse(textBoxEnmin.Text);
    Enmax = int.Parse(textBoxEnmax.Text);

    int rem;
    int count_En = Math.DivRem(Enmax - Enmin, En_diff, out rem);
    Es = double.Parse(textBoxEs.Text);
    N = int.Parse(textBoxN.Text);

    Fs = double.Parse(textBoxFs.Text);
    Cs = double.Parse(textBoxCs.Text);

    c = double.Parse(textBoxc.Text);

    COP = double.Parse(textBoxCOP.Text);

    ta = double.Parse(textBoxta.Text);

    UL = double.Parse(textBoxUL.Text);

    A = double.Parse(textBoxA.Text);

    Fm = double.Parse(textBoxFm.Text);

    Tp = Tgin = Tgout = double.Parse(textBoxTp.Text);

    Tlauko = double.Parse(textBoxTlauko.Text);

    Tv = double.Parse(textBoxTv.Text);

    list<double> listDeltaEnOnOff = new List<double>();
    listDeltaEnOnOff.Clear();

    m_EnOptimalus.Clear();

    int t;
    for (t = 0; t < N; t++)
    {
        m_EnOptimalus.Add(0);
    }

    int En_loc = Enmin;
    while (En_loc <= Enmax)
    {
        for (t = 0; t < N; t++)
        {

```

```

    m_EnOptimalus[t] = 0;
}

double Tvid = 0;
Calc_Hamiltonian(En_loc, out Tvid);

double old_hamiltonian = delta;
int candidate_j = -1;
short candidate_verte = 0;
int iteracijos = 0;
bool done = false;
while (!done)
{
    iteracijos++;
    done = true;
    for (t = 0; t < N; t++)
    {
        short prev_value = m_EnOptimalus[t];
        if (prev_value == 0)
            m_EnOptimalus[t] = 1;
        else
            m_EnOptimalus[t] = 0;

        Calc_Hamiltonian_Es(En_loc, out Tvid);

        if (delta < old_hamiltonian)
        {
            old_hamiltonian = delta;
            candidate_j = t;
            candidate_verte = m_EnOptimalus[t];
            done = false;
        }
        m_EnOptimalus[t] = prev_value;
    }
    if (!done)
        m_EnOptimalus[candidate_j] = candidate_verte;
}

Calc_Hamiltonian_Es(En_loc, out Tvid);
StringBuilder builder = new StringBuilder();
foreach (int EnValue in m_EnOptimalus)
{
    builder.Append(EnValue).Append(" ");
}
string rezimas = builder.ToString();

listEnConstOnOffEs.Add(new EnConstOnOffClass { En_galia = En_loc.ToString(), delta_suminis_integralinis_kvadratinis_nuokrypis = delta.ToString(),
Tvid_Temperatura = Tvid.ToString(), ReZimas = rezimas });
En_loc += En_diff;
}
double? minDelta = null;
double thisEn = 0;
string thisRezimas = "Neteisingai ivesties duomenys";
for (int i = 0; i < listEnConstOnOffEs.Count; i++)
{
    double thisDelta = Convert.ToDouble(listEnConstOnOffEs[i].delta_suminis_integralinis_kvadratinis_nuokrypis);
    if (!minDelta.HasValue || thisDelta < minDelta.Value)
    {
        minDelta = thisDelta;
        thisEn = Convert.ToDouble(listEnConstOnOffEs[i].En_galia);
        thisRezimas = listEnConstOnOffEs[i].ReZimas;
    }
}

textBoxMinDeltaEs.Text = minDelta.ToString();
textBoxEnOnOffEs.Text = thisEn.ToString();
textBoxRezimasEs.Text = thisRezimas;
}

private void buttonExcelEsKintanti_Click(object sender, EventArgs e)
{
    CreateExcelFile.CreateExcelDocument(listEnConstOnOffEs, "Galia nepastovi ON OFF Es kintanti.xlsx");
}

private void buttonExcelEsKintantiDabartine_Click(object sender, EventArgs e)
{
    CreateExcelFile.CreateExcelDocument(listEnConstOnOffDabartineEs, "Galia nepastovi ON OFF kas dt, Es kintanti.xlsx");
}

private void textBoxRezimasEs_TextChanged(object sender, EventArgs e)
{
}
}
}
}

```