

VILNIAUS UNIVERSITETAS

Ramūnas Dzindzalieta

JUDANČIŲ OBJEKTŲ STEBĖSENOS IR BŪKLĖS VERTINIMO
SISTEMOS IŠVYSTYMAS TAIKANT SESIJOS INICIJAVIMO
PROTOKOLĄ

Daktaro disertacija

Technologijos mokslai, informatikos inžinerija (07 T)

Vilnius, 2013

Disertacija rengta 2008 – 2013 metais Vilniaus universitete, Matematikos ir informatikos institute Programų sistemų inžinerijos skyriuje.

Mokslinis vadovas:

prof. dr. Dalė Dzemydienė (Vilniaus universitetas, technologijos mokslai, informatikos inžinerija – 07 T).

© R. Dzindzalieta,

2013

Reziumė

Disertacijoje nagrinėjamos sistemos, leidžiančios stebėti judančius objektus ir vykdyti bendradarbiavimą tarp skirtingų mobiliųjų įrenginių taikant vieningą duomenų apsikeitimo magistralę, kurios veikimas grindžiamas sesijos iniciavimo protokolo (SIP) funkcinėmis galimybėmis.

Atliekant judančių žemės paviršiumi objektų stebėseną (t.y. transporto priemonių, pavojingų krovinių vežimo) tenka nustatyti objekto buvimo vietą ir susieti kontekstinę informaciją su gaunamais duomenimis. Tam tikslui kuriamos judančių objektų stebėsenos ir būklės vertinimo sistemos. Tokios sprendimų paramos sistemos turėtų įgalinti bendrauti sensorinius nutolusius įrenginius, diagnozuoti bei vertinti susidariusių situacijų būklę ir spręsti pagalbos arba tinkamo reguliavimo uždavinius.

Tačiau, kuriant šias sistemas iškyla problemos susijusios su duomenų keitimusi tarp heterogeninių nutolusių įrenginių, perduodant sensorinės įrangos parametrų duomenis, sąveikios komunikacijos užtikrinimu, grįžtamojo ryšio valdymu ir sprendimo paramos modelio sukūrimu.

Pagrindinis tyrimo objektas – technologija, kuri užtikrina komunikaciją tarp heterogeninių nutolusių sistemų, praplečiant sesijos iniciavimo protokolo (SIP) funkcines galimybes taip, kad būtų galima stebėti, vertinti judantį objektą ir sąveikauti su kitais nutolusiais objektais.

Disertacijoje analizuojamos tokios mobiliųjų technologijų galimybės, kurios įgalintų platformą kuri leistų aptikti ir stebėti judančių objektų geografines koordinates bei fiksuoti sensorių informaciją nuotoliniu būdu.

Darbo tikslas yra išnagrinėti nutolusių objektų komunikavimo metodus ir pasiūlyti sprendimo paramos sistemos architektūrą, kuri integruotų sesijos iniciavimo protokolo išplėtotas funkcines galimybes nutolusių judančių objektų stebėsenai ir vertinimui, taikant vieningos duomenų mainų magistralės veikimo principus.

Darbe sprendžiami 4 pagrindiniai uždaviniai. Pirmas uždavinys skirtas nutolusių judančių objektų komunikacijos metodų analizei, taikant sesijos iniciavimo protokolą ir jo galimybių analizei, kad būtų galima užtikrinti sąveiką

lygiarangiuose (Peer-to-Peer), trečios 3G ir aukštesnės kartos tinkluose. Antras uždavinys skirtas sesijos inicijavimo protokolo (SIP) funkcinių savybių analizei, įgalinant realizuoti nutolusių objektų komunikavimą ir duomenų keitimąsi tarp heterogeninių mobilių įrenginių bei įvertinti jų komunikavimą. Trečias uždavinys skirtas nutolusių judančių objektų komunikacijos ir duomenų keitimosi įvertinimui, taikant Petri tinklų modeliavimo priemones. Ketvirtas uždavinys skirtas sprendimo paramos sistemos architektūros pasiūlymui, kuri integruotų sesijos iniciavimo protokolo išplėtotas funkcines galimybes, įgalintų vykdyti nutolusių objektų stebėseną ir vertinimą. Penktas uždavinys skirtas techninės ir programinės įrangos komplekto eksperimentiniam tyrimui atlikti, kad būtų galima įvertinti keturių skirtingų tipų nutolusių įrenginių apjungimą bendrai komunikacijai ir integracijai į SPS.

Disertaciją sudaro įvadas, penki skyriai ir išvados.

Įvadiniame skyriuje aptariama tiriamoji problema, darbo aktualumas, aprašomas tyrimų objektas, formuluojamas darbo tikslas bei uždaviniai, aprašoma tyrimų metodika, darbo mokslinis naujumas, darbo rezultatų praktinė reikšmė, ginamieji teiginiai. Įvado pabaigoje pristatomos disertacijos tema autoriaus paskelbtos publikacijos ir pranešimai konferencijose.

Pirmajame skyriuje apžvelgiamos judančių objektų (pasirinkti autotransporto objektai, judantys žemės paviršiumi) stebėsenos galimybės, aptariami kai kurie metodai, kurie leidžia taikyti mobiliąsias technologijas šių objektų geografinėi pozicijai nustatyti.

Antrajame skyriuje pateikiamas SIP veikimas, apžvelgtos galimybės šio protokolo integracijai, kad galėtų komunikuoti skirtingi nutolusieji įrenginiai. Pateikiamas SIP daugiasluoksnė architektūra ir aprašoma šio protokolo keturi sluoksniai, kurių kiekvienas vykdo atitinkamus funkcijų rinkinius. Apžvelgiama keturių skirtingų įrenginių tipų integravimo ir veikimo ypatybės bei galimybės sujungiant į tinklą.

Trečiajame skyriuje aprašoma spalvotųjų Petri tinklų pritaikymas SIP funkcinių savybių aprašymui, modeliavimui ir analizei. Pristatomos imitacinio modeliavimo priemonės – spalvotieji Petri tinklai (CPN) nutolusių mobilių

objektų stebėjimo scenarijams imituoti ir SIP veikimo procesams modeliuoti. Spalvotieji Petri tinklai leido stebėti sesijos inicijavimo protokolo skirtingų situacijų vykdomas būsenas konkrečiais laiko momentais ir duomenų srautų judėjimą, operacijų vykdymo sąlygas. Sukurti modeliai leido stebėti perduodamos informacijos struktūrą, kurią reikalinga įvertinti keičiantis SIP būsenoms, bei komunikavimo skirtingoms sąlygoms.

Ketvirtajame skyriuje pasiūlyta sprendimo paramos sistemos architektūra, kuri integravo sesijos inicijavimo protokolo išplėtotas funkcines galimybes.

Penktajame skyriuje aprašytas eksperimentas, kuris leidžia apjungti keturių skirtingų tipų įrenginius į visumą nutolusių įrenginių komunikacijai užtikrinti. Pristatomos techninės ir programinės įrangos komplektas, leidžiantis komunikuoti objektams tarpusavyje, praplėtus sesijos inicijavimo protokolo galimybes. Skyriuje pateikti eksperimentinių tyrimų rezultatai, kurių tikslas yra eksperimentiškai patvirtinti magistralės panaudojimą duomenų apsikeitimui tarp nutolusių įrenginių, diagnozuojant pavojingų objektų būsenas ir suteikiant atgalinį ryšį jų valdymui.

Turinys

ĮVADAS.....	14
1. METODAI IR PRIEMONĖS JUDANČIŲ OBJEKTŲ KOMUNIKACIJAI UŽTIKRINTI BELAIDŽIUOSE TINKLUOSE	23
1.1. Judančių autotransporto objektų samprata	23
1.2. Judančių objektų padėties nustatymo belaidžiuose tinkluose metodų apžvalga.....	26
1.3. Lygiarangių prietaisų sąveikos (Peer-to-Peer) tinklai	28
1.4. Sesijos inicijavimo protokolo taikymai P2P tinkluose.....	31
1.5. SIP lokacijos paslaugos išplėtimas.....	35
1.6. Judančių objektų SIP komunikacijos galimybės 3GPP tinkle	36
1.7. Pirmojo skyriaus išvados.....	41
2. DUOMENŲ KEITIMASIS TARP NUTOLUSIŲ HETEROGENINIŲ SISTEMŲ TAIKANT SESIJOS INICIJAVIMO PROTOKOLO GALIMYBES	42
2.1. SIP veikimo aprašymas	42
2.2. Sesijos inicijavimo protokolo sluoksniai ir būsenos	48
2.3. Vieningos komunikacijos magistralės sprendimai	53
2.4. Keturių nutolusių įrenginių tipų integracijos SIP vieningoje komunikavimo magistralėje sprendimai	55
2.5. SIP protokolo panaudojimas vieningoje magistralėje	59
2.6. Žiniatinklio paslaugų integravimo galimybės mobiliuosiuose įrenginiuose	60
2.7. SIP programavimo sąsaja mobiliems įrenginiams	63
2.8. Antrojo skyriaus išvados	64

3. PETRI TINKLŲ TAIKYMAS SIP KOMUNIKACIJOS PROCESAMS APRAŠYTI IR MODELIUOTI.....	66
3.1. Petri tinklų formalus aprašymas	66
3.2. Spalvotieji Petri tinklai (CPN).....	71
3.3. SIP veiklos procesų modeliavimo ir imitavimo galimybės taikant Spalvotuosius Petri tinklus (CPN)	73
3.4. INVITE transakcijos modeliavimas CPN priemonėmis	77
3.5. Sesijos inicijavimo protokolo laiko sąryšių modeliavimas taikant CPN	78
3.6. Sesijos inicijavimo protokolo kliento pusės transakcijų sekos aprašymas spalvotaisiais Petri tinklais.....	80
3.7. SIP serverio transakcijų modeliavimas	82
3.8. Galimos sesijos inicijavimo protokolo skambučių būsenos.....	83
3.9. Laikmačio B verčių keitimasis	84
3.10. Judančių objektų komunikacijos imitavimo privalumai taikant spalvotuosius Petri tinklus	85
3.11. Mobiliųjų objektų komunikacijos ir būsenos modeliavimas bei valdymas taikant spalvotuosius Petri tinklus	87
3.12. SIP programavimo sąsaja mobiliesiems įrenginiams ir sesijų valdymas	89
3.13. Trečiojo skyriaus išvados	91
4. SPRENDIMŲ PARAMOS SISTEMOS ARCHITEKTŪRA PAVOJINGŲ KROVINIŲ STEBĖSENAI IR BŪKLĖS VERTINIMUI TAIKANT MOBILIĄSIAS TECHNOLOGIJAS	92
4.1. Sprendimų paramos sistemos judančių objektų stebėsenai samprata	92
4.2. Sprendimo paramos sistemos pagrindinių komponentų architektūros aprašas	96

4.3. Kelio ruožų rizikos ypatumų aprašymas SPS sistemoje	99
4.4. Transporto objektų buvimo vietos duomenų atvaizdavimas, pasinaudojus mobiliomis technologijomis	101
4.5. Ketvirtojo skyriaus išvados	105
5. SIP INTEGRACINIŲ GALIMYBIŲ VERTINIMAS EKSPERIMENTIŠKAI APJUNGIANT SKIRTINGAS HETEROGENINES SISTEMAS	106
5.1. SIP išplėtimo eksperimentinio tyrimo sąlygų aprašas.....	106
5.2. Eksperimentinių tyrimų uždaviniai	107
5.3. Eksperimentinė platforma	107
5.4. Eksperimentinių tyrimų rezultatai	115
5.5. Eksperimentinės sistemos SIP žinutes apdorojančių įrenginių apkrovos tyrimas.....	116
5.6. Penkto Skyriaus išvados	120
IŠVADOS.....	122
Autoriaus publikacijų disertacijos tema sąrašas.....	137

PAVEIKSLŲ SĄRAŠAS

1.1 pav. Nestruktūrizuota belaidžių prietaisų sąveikos topologija, pagal (Dagher, 2008)	30
1.3 pav. SIP srauto nukreipimas tarpiniame serveryje	33
1.4 pav. Žinučių perdavimo iteratyvus mechanizmas pagal (Feng, 2011)	34
1.5 pav. Žinučių perdavimo rekursyvus mechanizmas pagal (Feng, 2011)	35
1.6 pav. SIP lokacijos serviso išplėtimas pagal (Shim, 2011)	35
1.7 pav. Mobilųjų įrenginių komunikacijos schema (Chu, 2012 ir Oredope, 2011)	39
1.8 pav. (šaltinis: Chu, 2012 ; Oredope, 2011)SIP registracijos žinutės perdavimo struktūra IMS architektūroje	40
2.0 pav. Sesijos inicijavimo protokolo taikomojo sluoksnio integracija su transporto ir tinklo sluoksniu (pagal IETF MMUSIC rekomendacijas).....	43
2.1 pav. Sesijos inicijavimo protokolo kliento pusės būsenų mašina (sudaryta autoriaus pagal IETF MMUSIC rekomendacijas).....	49
2.2 pav. Sesijos inicijavimo protokolo serverio pusės būsenų mašina (sudaryta autoriaus pagal IETF MMUSIC rekomendacijas).....	53
2.4 pav. Keturių įrenginių tipų nustatymo algoritmas.....	57
2.5 pav. Skirtingų tipų nutolusių įrenginių jungimo į bendrą magistralę, taikant SIP pavyzdys.....	58
2.6 pav. Duomenų keitimosi tarp dviejų mobiliųjų įrenginių schema taikant SOAP ir SIP	60
2.8 pav. SIP paslaugų praplėsto paketo klasių diagrama	63
3.1 pav. Sesijos inicijavimo protokolo imitacinis modelis CPN tinklais	78
3.2 pav. SIP klientinės pusės modelio CPN tinklas	82
3.3 pav. SIP serverio pusės imitacinis modelis CPN tinklais	83
3.4 pav. CPN schema, perduodamų duomenų paketus	89
3.5 pav. Bendra SIP veikimo struktūra, atvaizduojama CPN modeliu	90
4.2 pav. SPS pagrindinių komponentų architektūra	98
4.4 pav. Bendroji įrenginio ir serverio sąveikos komponentų architektūra	102
4.3 pav. Petri tinklo schema išreiškianti judančio objekto parametrų sekimo procesą	104
5.2 pav. SIP komunikacijos magistralės iliustracinė schema.....	109
5.3 pav. Aparatūrinės įrangos mikroschemos skirtos SIP valdymo pultui maketo pavyzdys	110
5.5 pav. Mikroschemos Atmega jungčių schema.....	111
5.7 pav. Mikroschemos Atmega tinklo jungčių schema	113
5.8 pav. Siūlomos sistemos komponentų sąveikos struktūra užtikrinant SIP komunikaciją.....	114
5.9 pav. Realiame laike stebimų žinučių siuntimo būklės stebėjimo ekranas ...	115
5.11 pav. SIP žinučių eilių dinamikos iliustracinis pavyzdys perkrautuose SIP serveriuose.	116
5.13 pav. SIP retransliuotų žinučių kiekio priklausomybė nuo laiko	120

LENTELIŲ SĄRAŠAS

1 lentelė.	Atsakų reikšmės į užklausas	49
2 lentelė.	SIP ir adapterio taikymo galimi įrenginių tipai ir jų pavyzdžiai.	54
3 lentelė.	SIP sąsajai užtikrinti reikalingos klasės	64
4 lentelė.	Srautą generuojančios komandos aprašas	119
5 lentelė.	SIP INVITE inicijuojama žinutės aprašas	119

SANTRUMPŲ IR TERMINŲ SĄRAŠAS

3G – telekomunikacijų standartas; 3G tinklas tai mobiliojo ryšio tinklas, kuris gali perduoti didelės spartos duomenis, teikiant interneto ir video telefonijos paslaugas.

Ad Hoc tinklas – (iš lotynų kalbos „tam tikslui“) decentralizuotos struktūros savaime susiorganizuojantis belaidis duomenų perdavimo tinklas.

API – (angl. Application Programming Interface) aplikacijos programavimo sąsaja, funkcijų ir bibliotekų rinkinys, kuris leidžia programai tiesiogiai bendrauti su operacine sistema.

Aplikacijų serveris – (IMS architektūroje) tai programinė platforma atsakinga už efektyvų paslaugų vykdymą ir saugojimą

CSCF – (angl. Call Session Control Server) skambinimo sesijos kontroliuojantis serveris.

CDMA – tai CDMA2000 ir cdmaOne standartų sutrumpinimas telekomunikacijose.

CDMA2000 – mobiliojo ryšio technologija, naudojama balso, informacijos ir signalų informacijos siuntimui tarp mobiliųjų telefonų ir stočių.

DIAMETER – kompiuterinių tinklų protokolas, skirtas prisijungimui, autorizacijai ir apskaitai (RADIUS protokolo pakaita).

DHT – (angl. Distributed Hash Table) iškirstytą maišos lentelė.

GATEWAY – tinklų sietuvas tinklo komponentas, siejantis tinklus, suderinantis skirtingus siejamų tinklų įvairių lygmenų protokolus, konvertuojantis (jei reikia) duomenis, persiųsdamas juos iš vieno tinklo į kitą.

GC – šiukšlių surinkiklis (angl. Garbage Collector).

GPRS – mobiliojo ryšio technologija, skirta duomenų perdavimui GSM tinkluose.

GPS – (angl. Global Positioning System) pasaulinė pozicionavimo sistema.

GSM – (angl. Global Standart for Mobile Communications) globalaus mobiliųjų telefonų ryšio standartas.

HSS – (angl. Home Subscriber Server) (IMS architektūra) IMS architektūros abonentų laikomų duomenų serveris, pagrindinė vartotojų duomenų bazė, kuri komunikuoja su IMS tinklo mazgais, kurie valdo skambučius.

IMS – (angl. IP Multimedia Subsystem) IP multimedijos posistemis, tai architektūra IP multimedijos paslaugoms pateikti.

IP – interneto protokolas.

JAIN – (angl. Java APIs for Integrated Networks) API aplikacijų interfeisų kūrimo Java aplinkoje integruotam tinklui programavimo kalba, tinkama kurti integruotas telefonijos, vaizdo, garso paslaugas.

Judantis objektas – darbe judantį objektą suprantame kaip žemės paviršiumi judančią transporto priemonę (automobilis, sunkvežimis ir pan.) ir keičiančią geografinę poziciją.

LAN – (angl. Local Area Network) lokalsios aprėpties tinklas.

MAC (angl. Media Access Control) – yra unikalus daugelio tinklo plokščių identifikatorius, kurį priskiria gamintojas.

OSI modelis – (angl. Open Systems Interconnection Reference Model) atvirų sistemų tarpusavio sąveikos modelis, naudojamas abstrakčiam kompiuterių tinklų ryšio protokolų specifikuojimui.

P2P (angl. Peer-to-Peer) – lygiarangių prietaisų komunikavimo technologija

Proxy serveris – tarpinis serveris.

SDP – (angl. Service Delivery Platform) paslaugų teikimo platforma, t.y. grupė komponentų (platforma) teikianti paslaugų valdymo architektūrą (jų sukūrimą, sesijos kontrolę, protokolus) tam tikro tipo paslaugoms.

SIP Servlets – SIP protokolo realizacija skirta telekomunikacijos paslaugų teikimui.

SIP – (angl. Session Initiation Protocol) sesijos inicijavimo protokolas dar kartais vadinamas signalizavimo protokolu, naudojamas multimedijos komunikavimo sesijų internete sukūrimui ir užbaigimui.

SPS – sprendimo paramos sistema (angl. Decision Support System).

TCP – (angl. Transmission Control Protocol) siuntimo valdymo protokolas, užtikrinantis patikimą duomenų perdavimą (OSI modelio 4 sluoksnyje).

UDP – (angl. User Datagram Protocol) duomenų perdavimo protokolas greitesnis nei TCP siuntimo protokolas, neatlieka duomenų tĕkmĕs kontrolĕs ir neturi klaidų atitaisymo mechanizmų, neuţtikrinantis patikimo perdavimo (OSI modelio 4 sluoksnyje).

VoIP – pokalbių perdavimas interneto protokolu.

WiFi – belaidžio tinklo ryŖio technologija.

WiMAX – (angl. Worldwide interoperability Microwave Access) belaidžio ryŖio sąveikumo technologija, uţtikrinanti mikrobangų prieigą, kuri leidžia sparĕiai perduoti duomenis dideliais atstumais.

WLAN – (angl. Wireless Local Area Network) belaidis lokalaus aprĕpties tinklas.

XML – (angl. Extensible Markup Language) duomenų struktūrų bei jų turinio apraŖomoji kalba.

P-CSCF – (angl. Proxy Call Session Control Server) nukreipiantysis skambinimo sesijos kontroliuojantis serveris

I-CSCF – (angl. Interrogating Call Session Control Server) apklausiantysis skambinimo sesijos kontroliuojantis serveris.

S-CSCF – (angl. Server Call Session Control Server) serverio skambinimo sesijos kontroliuojantis serveris.

RPC – (angl. Remote Procedure Call) nutolusios procedūros iŖkvietimas.

SOAP – (angl. Simple Object Access Protocol) paprastas objektų pasiekimo protokolas.

URI – (angl. Uniform Resource Identifiers) apraŖytas resurso identifikatorius.

WAN – (angl. Wide Area Network) globalios aprĕpties tinklas.

LAN – (angl. Local Area Network) vietinis aprĕpties tinklas.

ĮVADAS

Temos aktualumas.

Programinės įrangos kūrimas neretai tampa komplikuoatas, kai norima sujungti keletą heterogeninių sistemų į visumą bendrai komunikacijai užtikrinti. Informacinių technologijų vartotojai pageidauja vis naujų būdų ir galimybių keistis, apdoroti ir gauti reikiamus duomenis iš judančių objektų. Technologinių įrenginių spartus vystymasis verčia nuolat integruoti naujus įrenginius į jau egzistuojančias sistemas, nekeičiant visuminės sistemos architektūros, bet prijungiant naujas funkcines galimybes. Atsiranda poreikis projektuoti sistemą, kuri integruojama į jau esančią interneto telefonijos sistemą, išplečiant funkcines komunikacijos protokolo galimybes programinės įrangos dėka. Šiuolaikinės programinės įrangos kūrimas grindžiamas sparčiu interneto tinklų tobulėjimu. Įvairūs mobilieji įrenginiai, tokie kaip mobilieji telefonai, nešiojamieji kompiuteriai ir kiti specializuoti belaidžiai mechanizmai kartu su išvystytais komunikacijos tinklais populiarėja, suteikia galimybes bendrauti įrenginiais tarpusavyje, pasitelkus programinę įrangą. Šių technologijų išvystymas susiduria su išskirstytų sistemų darbu, jų sąveikos organizavimu, mobilių tinklų technologinių platformų taikymui. Išskirstytų sistemų dėka vartotojai gali efektyviau atlikti sudėtingas užduotis, bendradarbiauti ir koordinuoti savo veiksmus. Tokių sistemų privalumas – dalinimasis savo resursais ir gebėjimas nuskaityti parametrinius duomenis iš nutolusių heterogeninių sistemų.

Vystantis technologijoms, didėja skirtingų tipų įrenginių skaičius. Kiekvieno tipo įrenginiui dažniausiai kuriamos pagalbinės programos ar net sukuriamos technologinės platformos vien tam, kad jie galėtų tarpusavyje keistis duomenimis ir komunikuoti. Tai nėra visada patogiu ir tai aktuali problema vartotojams, kurie negali pasinaudoti kitoje platformoje teikiamu funkcionalumu. Šie vartotojai privalo turėti adapterius (techninę aparatūrą, kuri užtikrintų skirtingų įrenginių bendravimą). Tai nėra labai patogiu ypač tais atvejais, kai vartotojai kiekvienu atveju vis nauji ir kiekvienam įsigyti naują adapterį ne visada įmanoma.

Darbe pateikiamas pasiūlymas, kaip išvengti šių problemų nenaudojant skirtingų adapterių. Siūlomas būdas naudotis vieningu praplėstu komunikacijos protokolu, kuriuo galėtų naudotis, bet kuris vartotojas turintis mobilųjį įrenginį. Tokio sesijos iniciavimo protokolo su išplėstomis galimybėmis veikimo principas pagrįstas skambinimu, kai siunčiant kvietimą keistis duomenimis, pasiunčiama serveriui užklausa. Gavęs tokią užklausa, serveris neinicijuoja skambučio, o tiesiog vykdo jame įdiegtą papildomą funkcionalumą.

Plačiai paplitę esami infrastruktūriniai tinklai, kuriuose mobilieji įrenginiai jungiasi prie bazinių stočių per fiksuoto tinklo infrastruktūrą. Toks prisijungimo būdas yra grindžiamas centralizuota tinklo topologija, kurioje vartotojų duomenų persiuntimas vyksta per fiksuotą tinklo infrastruktūrą. Toks duomenų persiuntimas yra pakankamai lėtas ir ne visada tenkina poreikių. Todėl dabar pradėta vis plačiau naudotis internetiniu ryšiu. Pavyzdžiui, vartotojas prisijungia prie internetinių duomenų paslaugas teikiančių prieigos taškų ir turi greitesnes galimybes parsisiųsti duomenis. Prisijungimas prie šių prieigos taškų įgalina tiesiogiai perduoti garso ir vaizdo medžiagą.

Viena dažniausiai naudojamų technologijų tokiais atvejais yra GPRS, bet ši technologija yra labiau skirta duomenims pateikti, o ne tarpusavio komunikacijai užtikrinti.

Norint organizuoti sąveikumą tarp nutolusių įrenginių, tenka ieškoti būdų juos integruoti ir suteikti galimybes sujungti į vieną bendrą veikimo sistemą, taikant vieningą protokolą. Šiame darbe sprendžiama sesijos iniciavimo protokolo (SIP) galimybių integracijos problema, kad galėtų sąveikauti ir komunikuoti nutolę heterogeniniai įrenginiai ir tokiu būdu SIP taptų šių įrenginių komunikacijos vieningu protokolu. Informacinių sistemų vartotojams per sąsają, realiu laiku būtų pateikiami sensorių duomenys. Reikalui esant duomenys būtų pasiekiami belaidžiais įrenginiais (pvz.: mobiliuoju telefonu, planšete ar kt.). Visų nutolusių objektų komunikavimą sujungiant vieningu protokolu taip stengiamasi išvengti daugybės skirtingų dalykinių programų kūrimo, leidžiant nuskaityti ar valdyti nutolusius objektus.

Naudojantis šiuo principu darbe buvo apjungti keturių skirtingų tipų įrenginiai vieningu išplėstu protokolu, taip leidžiant jiems tarpusavyje komunikuoti. Yra ir kitų technologijų leidžiančių skirtingo tipo įrenginiams komunikuoti, bet naudojantis SIP protokolu mes taip pat galime integruoti naujus įrenginius į sistemą, juos apjungti ar net suteikti prieigą visiems fiksuoto ryšio įrenginiams.

Problemos formulavimas

Mobiliosios (belaidės) technologijos leidžia stebėti judančius objektus geografinėje vietovėje, tačiau sistemos, įgalinančios analizuoti objekto būsenas, reikalauja papildomų priemonių kūrimo ir integravimo į šių sistemų infrastruktūrą. Technologinė įranga, skaitanti duomenis iš sensorių, yra pagrindinis komponentas šių duomenų surinkimui ir duomenų saugyklas. Sensorinės įrangos (sensorių) atitinkamų parametrų reikšmės gali būti perduodamos ir nutolusius serverius, kuriuose šie duomenys yra sisteminami ir taikomi analizei bei informacijai dėl tolimesnių objekto valdymo veiksmų. Kadangi nutolę sensoriniai įrenginiai gali būti skirtingų tipų, dažnai iškyla problemų juos integruojant tarpusavyje. Integruojant daugybę skirtingų įrenginių į vieną sistemą, kyla suderinamumo problemos ir kiti nenumatyti techniniai sutrikimai. Dėl tos priežasties atsiranda papildomų darbo sąnaudų tas problemas išspręsti. Norėdami tinkamai valdyti nutolusius objektus, turime susieti kontekstinę informaciją su gaunamais duomenimis ir gaunamų duomenų pagalba diagnozuoti ir vertinti objekto būklę. Disertacijoje nagrinėjama vieninga duomenų apsikeitimo magistralė, grindžiama SIP protokolo veikimo galimybėmis. Iškyla klausimai, koku būdu judančių objektų duomenų apsikeitimą realizuoti, pasirenkant vieningą duomenų mainų magistralę ir sesijos iniciavimo protokolo funkcines galimybes.

Tyrimų objektas.

Tyrimų objektas – sprendimų paramos sistemos architektūra, kuri įgalina heterogeninių sistemų sąveikos užtikrinimą, integruojant sesijos inicijavimo protokolo funkcines galimybes ir leidžianti stebėti ir vertinti nutolusius judančius žemės paviršiumi objektus.

Disertacinio darbo tikslas

Išnagrinėti nutolusių objektų komunikavimo metodus ir pasiūlyti sprendimo paramos sistemos architektūrą, kuri integruotų sesijos inicijavimo protokolo išplėtotas funkcines galimybes nutolusių judančių objektų stebėsenai ir vertinimui, taikant vieningos duomenų mainų magistralės veikimo principus.

Darbo uždaviniai

1. Išnagrinėti metodus, kurie užtikrina nutolusių judančių objektų komunikaciją, taikant sesijos inicijavimo protokolą (SIP), nustatyti SIP galimybes užtikrinant sąveiką lygiaranguose (Peer-to-Peer) ir trečios kartos 3G tinkluose.
2. Išnagrinėti SIP funkcines savybes, kurios leidžia realizuoti nutolusių judančių objektų komunikaciją ir duomenų keitimąsi tarp heterogeninių nutolusių skirtingo tipo įrenginių.
3. Išnagrinėti metodus, leidžiančius modeliuoti ir imituoti sudėtingų sistemų komunikacijos procesus ir įvertinti sesijos iniciavimo protokolo savybes nutolusių judančių objektų komunikacijai ir duomenų keitimui užtikrinti pasinaudojus ir pritaikius spalvotųjų Petri tinklų modeliavimo priemones.
4. Pasiūlyti sprendimo paramos sistemos architektūrą, kurioje būtų integruotos SIP išplėtotos funkcinės galimybės, leidžiančios vykdyti nutolusių objektų stebėseną ir situacijos vertinimą.
5. Pasiūlyti techninę ir programinę įrangą, kuri leistų nutolusiems objektams komunikuoti tarpusavyje, naudojant SIP išplėtotas funkcines ir vieningos duomenų perdavimo magistralės galimybes bei eksperimentiškai įvertinti pasiūlytos techninės ir programinės įrangos tinkamumą skirtingų heterogeninių įrenginių komunikacijai.

Tyrimų metodai.

Darbe buvo panaudoti analitiniai palyginamieji metodai analizuojant naujausią mokslinę literatūrą iš mokslinių duomenų bazių. Išsiaiškinti kitų autorių pateikiami lygiarangių prietaisų sąveikos ir komunikavimo metodai, duomenų perdavimo magistralės funkcionavimo būdai. Analizuoti sesijos inicijavimo protokolo daugiasluoksnės architektūros sąveikos bei integravimo metodai.

Išanalizavus literatūrą, buvo pasiūlytas duomenų keitimosi tarp heterogeninių įrenginių būdas, šių įrenginių suderinimui bei susiejimui į bendrą duomenų mainų magistralę. Taikyti sesijos inicijavimo protokolo veiklos modeliai, grindžiami baigtinių automatų modeliavimo, konceptualaus modeliavimo spalvotųjų Petri tinklų modeliavimo priemonėmis. Tam tikslui sisteminės procesų analizės, sistemų veiklos procesų modeliavimo ir imitavimo metodai bei įrankiai (pvz.: CPN Tools, TCPDump, Wireshark, Hpink3). Pritaikyti duomenų apsikeitimo tarp heterogeninių įrenginių metodai, pasinaudojus SIP protokolo funkcinėmis galimybėmis.

Eksperimentiniam tyrimui atlikti sukurtas ir panaudotas prototipinis techninės įrangos maketas, kurį sudarė Atmega plokštė (mikrokontrolėlis), interneto lizdas, nutolę sensoriai, leidžiantis apjungti 4 skirtingų tipų nutolusius įrenginius, skirtingai komunikacijai užtikrinti, vieninga duomenų perdavimo magistralė ir kt., įgalinantis įvertinti komunikacinės sistemos darbą. Ekperimento rezultatams palyginti taikyti eilių teorijos metodai, kurie leido nustatyti serverio apkrovų, žinučių retransliacijos pajėgumus ir palyginti juos kiekybiškai.

Mokslinio darbo naujumas.

Rengiant disertaciją buvo gauti šie informatikos inžinerijos mokslui naudingi rezultatai:

- Atlikta tankiųjų tinklų, lokaliųjų aprėpties tinklų, 3G tinklų, proginių (ad-hoc) tinklų ir lygiarangių (Peer-to-Peer) tinklų galimybių analizė parodė, kad nutolusių įrenginių bendravimui naudojama didelė įvairovė komunikacijos protokolų, o skirtingiems įrenginiams kūriami vis nauji papildomi prietaisai (adapteriai), leidžiantys nutolusiems įrenginiams komunikuoti tarpusavyje esant skirtingiems protokolams. Todėl siūloma spręsti šią problemą praplečiant SIP galimybes ir įgalinant komunikuoti heterogeninius nutolusius įrenginius tokių būdu, kad SIP taptų šių įrenginių komunikacijos vieningu protokolu.
- Išnagrinėtos sudėtingų dinaminių sistemų modeliavimo ir imitavimo priemonės (Petri tinklai, Spalvuotieji Petri tinklai, baigtiniai automatai ir kt.) leido įvertinti spalvuotuosius Petri tinklus kai vieną šių sistemų

modeliavimo ir imitavimo priemonių. Spalvotuoju Petri tinklų pagrindu sudarytas SIP protokolo daugialygis modelis sudarytas SIP protokolo veikimo modelis leidžia ištirti atliekamas operacijas ir stebėti imitaciniame modelyje vykdomus procesus ir būsenas. Šis SIP protokolo veikimo modelis leido išanalizuoti ryšių kanalais perduodamos informacijos struktūrą, kurią reikalinga įvertinti keičiantis protokolo būsenoms, bei matyti, kaip yra atliekami valdančios informacijos perdavimai ir informacijos srautų judėjimai.

- Pasiūlyta sprendimo paramos sistemos architektūra, kuri integravo tradicinius SPS sistemos komponentus ir mobilių technologijų pagrindu kuriamos įterptines posistemes (nutolusių sensorių įranga ir kt.), kuri integravo ir leido apjungti nutolusius skirtingo tipo įrenginius SIP pagrindu, atsisakant papildomų adapterių. Tai suteikė galimybių integruoti naujus sensorinius įrenginius į sistemą, nedarant pakeitimų bendroje architektūroje. Įrenginiai buvo suklasifikuoti pagal jų veikimą į keturis tipus ir nustatytos techninės ir programinės jų komunikacijos sąlygos pagal SIP galimybes užmegzti tarpusavio sesijas. Išplėtotą sistemos architektūrą leido apjungti skirtingus įrenginius į visumą, jų tarpusavio komunikacijai pasirinkus SIP, kaip pagrindinį protokolą.

Darbo praktinė svarba.

SIP protokolo žinučių nuskaitymas ir sesijų inicijavimas tarp SIP protokolą palaikančių įrenginių ir SIP adapterio sukūrimas įrenginiams, praktiškai galės būti pritaikomas ir tiems, kurie nepalaiko SIP protokolo ir nėra IP tinkle. Pasiūlytas nuotolinės sensorinės objektų stebėsenos sistemos prototipinės architektūros modelis. Šis modelis leidžia integruoti sensorines komponentes palaikančias ryšį tarp skirtingų tipų mobiliųjų įrenginių, perduodant sensorių užfiksuotus duomenis ir įvertinant judančio objekto būklę.

Naudojami metodai ir programinė įranga belaidėse sistemose leidžia keistis dauguma įmanomų duomenų formatų (pvz., tekstiniais, balsiniais ar vaizdiniais), suteikia informacijos apie objekto būseną realiuoju laiku. Pagrindinis privalumas

– išplečiama architektūra, kad galėtume gauti duomenis mobiliuoju telefonu per interneto paslaugas.

Autorius tiesiogiai prisidėjo prie projekto, vykdomo TEO LT ir vykdė darbus, susijusius su SIP protokolo galimybių išvystymu. Ši sistema inicijuoja ir valdo skambučius. Plačiau su šios sistemos galimybėmis galima susipažinti interneto svetainėje¹.

Pasiūlyta SPS sistemos architektūra įgalins pereiti prie belaidžių įrenginių komunikavimo ir informacijos vertinimo, taikant SIP išplėtimo galimybes. Šiame tiriamajame darbe yra sprendžiamos problemos, susijusios su ryšio palaikymu tarp skirtingų įrenginių, pasinaudojus vieningu protokolu, šiuo atveju – sesijos inicijavimo protokolu.

Ginamieji teiginiai.

- Pasiūlant sprendimo paramos sistemos architektūrą, kuri leistų stebėti judančių žemės paviršiumi objektus ir vertinti jų būklę, tikslinga SPS sistemą projektuoti kaip tradicinę, integruojant įterptines posistemas, kurios užtikrintų judančių sensorių stebėseną, duomenų saugojimą nutolusiuose duomenų saugyklose ir komunikaciją taikant vieningą SIP protokolą sesijos užmezgimui, palaikymui ir nutraukimui.
- Sesijos inicijavimo protokolo (SIP) architektūrinių sprendimų galimybių analizei atlikti tikslinga taikyti modeliavimo ir imitavimo priemonę - Spalvotuosius Petri tinklus. Sukurti modeliai leidžia išanalizuoti ryšių kanalais perduodamos informacijos struktūrą, kuri reikalinga keičiant vieną būseną kita, bei matyti, kaip jais atliekami valdančios informacijos perdavimai ir informacijos srautų judėjimai.
- Heterogeninių nutolusių sistemų komunikacijai užtikrinti lygiaranguose (Peer-to-Peer), 3G ir aukštesniuose tinkluose tikslinga suklasifikuoti įrenginius į keturis tipus ir jų komunikacijos valdymui taikyti SIP išplėtotas galimybes.

¹ http://www.teoverslas.lt/pokalbiai/paslaugos/greitasis_skambutis

- Pasiūlyta techninė ir programinės įranga, leidžia užtikrinti ir išbandyti nutolusių objektų komunikavimą ir integruoti bei įvertinti SIP praplėstas funkcines ir vieningos duomenų perdavimo magistralės galimybes, bei pagerina SIP vieningos magistralės funkcijas.

Darbo rezultatų aprobavimas.

Disertacinio darbo rezultatai pristatyti 6 tarptautinėse konferencijose ir 3 nacionalinėse konferencijose.

Tarptautinėse konferencijose:

1. 13th East-European Conference on Advances in Databases and Information Systems September 7-10, 2009 Riga, Latvia
2. EURO Mini Conference 5th International Vilnius Conference on Sustainable Development “Knowledge-Based Technologies and OR methodologies for Strategic Decisions of Sustainable Development”, September 30–October 3, 2009, Vilnius
3. International Conference "Reliability and Statistics in Transportation and Communication" (RelStat'10), 20-23 October 2010, Riga, Latvia.
4. International Conference „Social technologies ’11: ICT for Social Transformations“, Vilnius-net, November 17-18, 2011, Vilnius
5. Tenth International Baltic Conference on Databases and Information Systems (Baltic DB&IS 2012), July 8-11, 2012, Vilnius, Lithuania (2 pranešimai).

Nacionalinėse konferencijose:

6. XIII tarptautinė mokslinė Kompiuterininkų konferencija „Kompiuterininkų dienos“. 2009, Kaunas
7. Technologijos mokslo darbai Vakarų Lietuvoje, 2010, Klaipėda, Klaipėdos universitetas
8. 3-ioji Lietuvos jaunųjų mokslininkų konferencija „Operacijų tyrimai verslui ir socialiniams procesams“ 2010 m. spalio mėn. 1 d., kurią organizavo Lietuvos operacijų tyrimų draugija, Matematikos ir

informatikos institutas, Mykolo Romerio universitetas, Vilniaus Gedimino technikos universitetas. Vilnius.

Darbo tematika paskelbtos 13 publikacijų. Praktiniai darbo rezultatai taikomi dirbant telekomunikacijų bendrovėje TEO LT, dirbant tinklo plėtros valdymo departamente.

Darbo apimtis ir struktūra

Disertaciją sudaro įvadas, 5 skyriai, išvados, naudotos literatūros sąrašas, autoriaus publikacijų sąrašas.

Disertacijos apimtis 139 puslapiai, iš jų 5 lentelės ir 35 paveikslėliai.

1. METODAI IR PRIEMONĖS JUDANČIŲ OBJEKTŲ KOMUNIKACIJAI UŽTIKRINTI BELAIDŽIUOSE TINKLUOSE

Šiame skyriuje apžvelgiami metodai ir priemonės, kurie leidžia užtikrinti judančių objektų komunikaciją belaidžiuose tinkluose. Apžvelgiamos judančių objektų (pasirinkti autotransporto objektai, judantys žemės paviršiumi) stebėsenos galimybės, aptariami kai kurie metodai, kurie leidžia taikyti mobiliąsias technologijas. Svarstoma, kokios kylančios problemos gali įtakoti operatyvių sprendimų priėmimus ir nagrinėjamos jų sąveikavimo aprašymo galimybės. Autorius pateikia svarbiausius mokslinėje literatūroje aprašomų tinklų komunikacijos būdus ir apžvelgia SIP integracijos galimybes šiuose tinkluose. Skyriuje apžvelgiami Sesijos inicijavimo protokolo taikomieji pavyzdžiai P2P, lokaliuose 3G ir aukštesnės kartos tinkluose. Pateikiama SIP integracija IMS architektūroje.

1.1. Judančių autotransporto objektų samprata

Transporto srautų ir avarijų keliuose didėjimas verčia nagrinėti krovinių transportavimo problemą sisteminiu daugiakompleksiniu požiūriu.

Judančių transporto objektą šiame darbe suprasime kaip judančią žemės paviršiumi transporto priemonę. Tokių transporto priemonių, vežančių pavojingus krovinius, srautai didėja, ir kiekviena iš jų tampa galimos pavojingos situacijos potencialiu dalyviu (Ruifang, 2010). Todėl darbo aktualumas pasireiškia tuo, kad bus nustatytos galimos rizikos sritys ir pasiūlyti operatyvaus valdymo metodai.

Transportuojant krovinius, vežančius pavojingas, taršias medžiagas, išauga aplinkos taršos rizika, susijusi su judančių transporto objektų įvykių pasireiškimo tikimybinio padidėjimu ir atitinkamų pasekmių galimybėmis. Tikėtinos taršos rizika įgyja daugiafunkcinį charakterį ir yra sudėtingesnė nei rizikos vertinimas stacionarių objektų aplinkoje (Arpaia, 2005). Darnios aplinkos vystymo reikalavimai įpareigoja užtikrinti saugius transportavimo procesus ir ypač saugų pavojingų krovinių transportavimą skirtingose autotransporto priemonėse ir teritorijose. Todėl gana svarbu sukurti tinkamą, sprendimus palaikančią aplinką,

kuri padėtų stebėti ir operatyviai vertinti transporto objektus, vežamų cheminių medžiagų būklę, kelio ruožų avaringumą. Ypatinę reikšmę įgyja transportavimo kelyje išsidėsčiusių teritorijų statusas ir padėtis. Šiam tikslui tenka ieškoti metodų, tinkamų vertinti pavojingų krovinių transportavimo procesus ir juos aprašyti sprendimų paramos sistemoje (SPS).

Mobilios technologijos padeda nustatyti objekto buvimo vietos informaciją, leisti sieti ją su vietovės informacija, taikyti atitinkamus sensorių parodymus. Vietovės nustatymas grindžiamas priminimų paslauga ir veikia belaidžiuose įrenginiuose. Priminimai gali būti naudingi, kai kontekstinė informacija pateikiama tinkamu metu ir tinkamoje vietoje. Prietaisų sensoriai sąveikauja su fizine aplinka per sensorius (Peterkin, 2011). Pavojingų krovinių gabenimo kelias ilgas ir kiekviena akimirka keičiasi nuo daugelio faktorių, todėl pavojingų krovinių vežimui keliami ypatingi reikalavimai ne tik šalies, bet ir tarptautiniu mastu. Nėra vieningos sistemos, kuri koordinuotų pavojingus krovinius (Valente, 2009). Nėra sudarytos informavimo sistemos apie pavojingų krovinių būklę kiekvienu laiko momentu, kas leistų vairuotojui priimti reikiamas išvadas.

Per Lietuvą kasdien vežama daugybė tonų pavojingų krovinių, kurie gali išsilieti, sprogti ar kitaip paveikti žmonių sveikatą, turtą ar aplinką. Pagrindinės incidentų priežastys yra šios:

- Transportuojančių krovinių vežimo reikalavimų nesilaikymas. Nėra tinkamos kontrolės arba šios kontrolės nesilaikoma, arba kontrolė yra neveiksminga.
- Teikiamų darbų tikslas yra teoriškai ištirti ir parengti pasiūlymus, kaip efektyviai organizuoti pavojingų krovinių vežimą atsiradus nenumatytoms situacijoms ir dinamiškai keisti pasiūlymus užsakovui.
- Rizikos kontrolė yra svarbu rizikos valdymui. Pavojingų medžiagų išmetimas į aplinką avarijos momentu yra rimta pasekmė. Pavojingų krovinių avaringumas priklauso nuo kelio charakteristikos ir transporto priemonės tipo. Tačiau kelių ir transporto priemonių savybes sunku pakeisti transportavimui.

- Rizikos valdymas – tai struktūrizuotas, suderintas ir nenutrūkstamas procesas, padedantis nustatyti ir įvertinti galimybes ir pavojus, turinčius įtakos tikslams pasiekti, taip pat leidžiantis priimti sprendimus, atsižvelgiant į esamus veiksmus. Rizikos valdymas padeda pasiekti geresnių rezultatų ir iš esmės sumažina nuostolių patyrimo galimybę. Rizikos suvaldymas turi būtų nepetrūkstantis procesas, glaudžiai susijęs su pavojingo transporto duomenų nuskaitymu ir interpretavimu.

Rizikos analizė rodo, kad pavojingų medžiagų gabenimas yra sudėtingas procesas ir sukelia visai kitokius pavojus nei statiniai objektai.

Aplinkos tvarumas priklauso nuo saugaus transportavimo, ypač pavojingų krovinių saugaus gabenimo skirtingomis transportavimo rūšimis (Qiulin, 2010).

Mobiliosios technologijos potencialiai patogi ir visa apimanti platforma, skirta nustatyti objekto kontekstinę informaciją, tokią kaip vietovės pobūdis ir šios informacijos pateikimas esant atitinkamoje geografinėje pozicijoje apgyvendintos vietovės žmonių skaičius susijęs su pavojingo krovinio gabenimo informacija.

Priminimai naudingi tada, kai pavojingo transporto kontekstinę informaciją būtina pateikti atitinkamais laiko momentais ir atitinkamose vietose.

Prietaisai – sensoriai, jutikliai, naudojami tam tikros fizinės aplinkos informacijos nuskaitymui. Bendrais bruožais, bet koks įrenginys, kuris leidžia jausti ir bendrauti su fizikine aplinka, gali būti laikomas jutikliu.

Mobiliosios technologijos leidžia šiuolaikinei platformai aptikti ir stebėti judančio transporto geografines koordinates. Atitinkamų parametrų reikšmės gali būti fiksuojamos sensorinės įrangos sensorių priemonėmis ir perteikiamos nuotoliniu būdu į centrinę duomenų bazę. Norint tinkamai valdyti transporto priemones, tenka ne tik nustatyti objekto buvimo vietą, bet ir susieti kontekstinę informaciją su gaunamais duomenimis ir diagnozuoti, vertinant susidariusios situacijos būklę (Valente, 2009). Taip pat sudaryti sąlygas pasiekti transporto priemonę tam tikroms tarnyboms, sprendžiant pagalbos arba tinkamo reguliavimo uždavinius.

1.2. Judančių objektų padėties nustatymo belaidžiuose tinkluose metodų apžvalga

Kiekvieno judančio objekto įrenginio vieta gali būti nustatyta GPS (Global Positioning System) imtuvo pagalba. GPS koordinatės persiunčiamos į centrinį serverį, kurios reikalingos tolesnei sistemos vartotojo analizei ir nustatyto tipo ataskaitų generavimui (Herrera, 2009). Vartotojai, norintys pamatyti judančio objekto koordinatės ar kitus teikiamus duomenis, jungiasi prie centrinio serverio, kuriame mato tik tam tikro laiko duomenis, kurie buvo gauti.

Veikiant GPS imtuvui naudojama pakankamai daug elektros energijos, todėl juos pastoviai naudoti yra neekonomiška, neturint pastovaus maitinimo šaltinio. GPS imtuvo įjungimas ir vietos nustatymas priklausomai nuo modelio gali užtrukti nuo vienos iki kelių minučių, todėl reikia nuspręsti kiek dažnai vietos geografinė pozicija bus gaunama įrenginyje.

Šiai problemai spręsti siūlomi būdai:

- GPS imtuvas įjungiamas periodiškai tam tikru laiko intervalu, nuskaitinėjant ir įrašant duomenis į atmintį, objektui judant tam tikru greičiu v .
- Jei judantis objektas nejuda, jo greitis $v=0$, geografinė pozicija nuskaitinėjama rečiau.
- GPS imtuvas yra išjungiamas tol, kol nėra poreikio atnaujinti vietos informaciją.
- GPS imtuvas yra išjungiamas, kai objekto judėjimo greitis yra $v>0$, su sąlyga, kad judantis objektas aptiko prieigos tašką, galėdamas prisiregistruoti SIP žinute, taip gaudamo to prieigos taško koordinatės ir priskiriamos prieigos taško geografinės pozicijos taško koordinatės kaip savo.

Vietose kur nėra tiesioginio matomumo su palydovais ir GPS imtuvai neveikia (pvz. tai galėtų būti uždaros patalpos, tuneliai ir kt.) reikalingos kitos priemonės vietos nustatymui.

Tokiose vietose kur neveikia GPS ryšys naudojama belaidžio ryšio infrastruktūra, susidedanti iš prieigos taškų buvimo vietai nustatyti.

Vartotojai su savo įrenginiais jungiasi belaidžiu ryšiu į bendrą tinklą per prieigos taškus - PT (angl. access point). Taip sujungti vartotojų tinklo įrenginiai sudaro belaidžius lokalsios aprėpties tinklus (angl. Wireless Local Area Network - WLAN). Net būdami arti vienas kito, vartotojo įrenginiai prisijungia prie prieigos taškų ir siunčia duomenis tik per juos (Plestys, 2009, Andziulis, 2012).

Pradžioje mobilusis įrenginys, norėdamas prisijungti prie lokalsios aprėpties tinklo, išsiunčia registracijos žinutę, kurioje nurodo savo būvimo vietą bei kitus duomenis, leidžiančius užmegzti sesiją. Kreiptis į kitos rūšies tinklus ir naudotis internetu iš lokalsios aprėpties tinklų galima per prieigos taškus.

Vietovėse, kuriose reikalaujama išlaikyti didelio ryšio kanalų pralaidumą ir fiksuoto tinklo įrengimas yra per brangus, diegiami tankieji tinklai, kurie pasižymi ribotu mobilumu, todėl jie naudojami stacionarioms belaidžio tinklo paslaugoms teikti. Tankiojo tinklo maršrutizatoriai naudoja skirtingus dažnių ruožus ir turi keletą belaidžio ryšio modulių. Tankiuosiuose tinkluose naudojami integruotų technologijų standartų (Ghazisaidi, 2009), pvz., IEEE 802.16 (WiMAX) (IEEE 802.16-2009) įrenginiai. Tai leidžia apjungti atskirus IEEE 802.11a/b/g/n standartų tinklus suteikiant didelį ryšio kanalų pralaidumą.

Padėties paslaugos (angl. location services) yra skirtos įrenginių vietos skleidimui tinkle. Išskiriamos dvi pagrindinės šios paslaugos rūšys (Lee, 2002)- padėties duomenų bazių sistemos (angl. location database systems) ir padėties informacijos skleidimo sistemos (angl. location dissemination systems). Tinklo įrenginiai perduoda padėties koordinačių atnaujinimo pranešimus duomenų bazės serveriui. Vietos nustatymo paslaugų duomenų bazės yra kopijuojamos, norint padidinti sėkmingo atsakymo į užklausą tikimybę. Vienas iš tokių sistemų pavyzdžių yra tinklinė vietos nustatymo paslauga Grid Location Service (GLS) (Pleštys, 2010)

Straipsnyje (Acker, 2010) pristatė namo valdymo sampratą naudojančią VoIP media posistemės privalumus, kuriais valdomi įrenginiai naudojami KNX magistrale. KNX magistralė apjungia įvairias prietaisų valdymo technologijas: magistralinio kabelio, jėgos kabelio, interneto, infraraudonųjų spindulių, radijo

bangų. Namų valdymas pagrįstas KNX technologijos naudojimu. Tai leidžia perduoti duomenis internetiniu protokolu. (Acker, 2010) straipsnyje akcentuojama, kad nutolusių įrenginių valdymas stipriai prisideda prie aplinkos tausojimo, mažindamas energijos suvartojimą. (Kim, 2007) straipsnyje pasiūlytas duomenų nuskaitymas ir jų pristatymas vartotojui, pasinaudojant UDP protokolu, taip pat įrenginių ir vartotojo komunikacijai užtikrinti buvo pasirinktas SIP protokolas. Kim pasiūlė monitoringo sistemą sąveikos stebėsenai ir duomenų perdavimui realiuoju laiku, panaudojus SIP protokolą ir ZIGBEE tinklo integraciją. (Jung, 2008) straipsnyje pateikta ZIGBEE įrenginių integracija duomenų siuntimui, pasinaudojus GPRS ryšiu. Kadangi tinkle gali būti keletas ZIGBEE įrenginių, tai kad jie teisingai pristatytų duomenis, buvo įvestas identifikatorius.

1.3. Lygiarangių prietaisų sąveikos (Peer-to-Peer) tinklai

Lygiarangių prietaisų sąveikos (angl., Peer-to-Peer) tinklai, sutrumpintai vadinami P2P tinklais plėtojosi nuo kliento – serverio paradigmos (Li, 2008). Šios paradigmos pagrindas yra tas kad klientai gali naudotis nutolusiais serverio ištekliais. Tačiau kliento – serverio architektūriniai sprendimai turi keletą trūkumų, tokių kaip kliento pusės plėtojimo galimybės (scalability), vieno „taško“ gedimą serverio pusėje ir neišnaudotų resursų galimybes tinkle.

Belaidžių technologijų vystymas reikalingas sudaryti palankesnes sąlygas daugiakomponentinių funkcijų atlikimui, pateikiant naujas paslaugų atlikimo galimybes, kurios būtų prieinamos vartotojui bet kuriuo metu, nepriklausomai nuo vietos. Šių technologijų galimybės grindžiamos atvirųjų paslaugų platforma ir diegiama mobiliams vartotojams.

Mobiliųjų technologijų sklaidos mechanizmai įgalina pasinaudojus belaidžiu tinklu susisiekti su išorinėmis sistemomis. Mobiliųjų technologijų pasiekimai daro įtaką aukšto lygio naujų duomenų pateikimo formų skvarbumui (angl. penetration), paslaugų išdalinimui (angl. granularity) ir individualiai, asmeniui orientuotai aplinkai (Chang, 2011)

Sensoriniai įrenginiai, nutolusius įrenginius integruojant į bendrą tinklą, grindžiamą SIP protokolu, smarkiai pakeis informacinių sistemų ir telekomunikacijų rinką (Qin, 2010; Zhou, 2010). Šios srities ekspertai prognozuoja technologijos proveržį artimiausiais metais. Proveržis turėtų būti paskatintas mažėjančiomis komponentų kainomis ir platesniu technologijų pasiekiamumu (Bertran, 2009). Jau dabar pasaulyje yra milijardai įrenginių, kurie galėtų būti sujungti į tinklą antžemiais arba belaidžiais telekomunikacijų tinklais. Dėl mažėjančių telekomunikacijos įrenginių kainų vis daugiau jų bus sujungiami. Duomenys, surinkti iš sensorinių įrenginių, pagelbės ne tik jų vartotojams, bet ir pakeis nemažą dalį dabartinių darbo principų.

Šiuo metu sensorinių įrenginių technologijos yra pasiekiamos platesnei rinkai. Technologijų plėtra, mobiliųjų tinklų plėtra ir duomenų perdavimo greičio augimas skatina belaidžių paslaugų plėtrą, kai belaidis įrenginys gali bendrauti su kitu belaidžiu įrenginiu. Ši technologija jau tapo prieinama ne vien verslui, bet ir namų ūkių privačiam naudojimui.

Mobiliojo ryšio technologijų plėtra taip pat prisideda prie naujų technologijų pritaikymo. Pastaraisiais metais plačiai paplitus 3G technologijai, atsirandant 4G ir plečiantis WiMax tinklui atsirado naujų galimybių – mobiliuoju ryšiu transliuoti vaizdą ir garsą, bei palaikyti pastovų duomenų ryšį. Naujos galimybės reiškia ir naujas paslaugas galutiniam vartotojui.

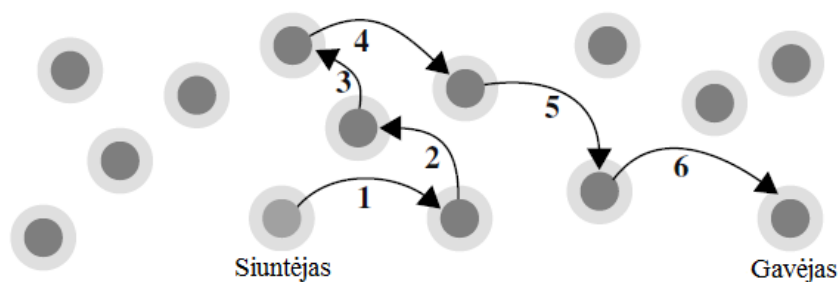
Paskutiniaisiais metais kompiuterių lustai tampa vis mažesni ir mažesni. Gamintojai juos pritaikė ne tik kompiuteriams, bet ir mažiems įrenginiams, tokiems kaip mobilieji telefonai, delniniai kompiuteriai, elektroninės užrašų knygutės, žiniatinklių telefonai ir televizijos priedėliai (set of box). Pagrindinė kompiuterių nauda yra ta, kad efektyviau išplečia aplinką mūsų versle ir asmeniniame gyvenime, leidžia naudotis kompiuterių galimybėmis, nepriklausomai nuo buvimo vietos. Įrenginių, pritaikytų mobiliosioms technologijoms, yra daug, ir jie gali būti naudojami įvairiausiems tikslams. Tai gali būti, pavyzdžiui, nešiojamieji ir PDA kompiuteriai, mobilieji telefonai ir specialūs pranešimų gavikliai. Mobiliosiomis technologijomis gali būti valdoma

ir naudojama informacija, esanti kitame mieste, nutolusiuose serveriuose arba interneto svetainėse bei sistemose.

Mobilusis prietaisas gali būti suprantamas kaip mažas, nešiojamas įrenginys (tai gali būti kompiuteris, mobilus telefonas ar pan.), kuriame galima saugoti, valdyti bei gauti informaciją nuotoliniu būdu belaidėmis technologijomis. Mobilieji prietaisai gali veikti skirtingose operacinėse sistemose, tokiose kaip *Android*, *Linux* ar supaprastintose *MS Windows* versijose arba tose, kurios skirtos būtent mobiliems prietaisams. P2P tinkluose prietaisai yra lygiarangiai (peers), t.y., jie turi kliento ir serverio vaidmenis tuo pačiu metu (Meyer, 2008). Tokiu būdu dalies P2P sistemos gedimas neturės sunkių pasekmių visai sistemai. Kitos sistemos dalys aplenks sugedusias dalis.

P2P sistemos yra suskirstytos į dvi grupes: hibridines ir grynąsias P2P sistemas. Kol P2P sistemos turi kliento - serverio komponentus, grynos P2P sistemos neturi jokio vieno „taško“ gedimo dėl kliento - serverio komponentų. Yra du grynujų P2P sistemų tipai paskirstytieji ir lokalizuotieji sistemos viduje.

Nestruktūrizuoto P2P tinklo topologijoje (1.1 pav.) yra „užlieti“ tinklą paskleistomis žinutėmis artimiausių galimų gavėjų rate, t.y. skleisti žinutes nuo vieno lygiarangio prietaiso iki kito. Žinutė yra persiunčiama kiekvienam žinomam kaimyniniam lygiarangiam prietaisui, kol žinutė pasiekia tikslą. Viena iš populiarių P2P sistemų, kuri naudoja nestruktūrizuotą topologiją, yra Gnutella (Ripeanu, 2001).



1.1 pav. Nestruktūrizuota belaidžių prietaisų sąveikos topologija, pagal (Dagher, 2008)

Nestruktūrizuotos P2P sistemos naudoja daug tinklo išteklių, kad skleistų žinutes visiems žinomiems kaimyniniams lygiarangiems prietaisams (Dagher,

2008). Šis faktas paskatino P2P sistemų plėtrą. Šios sistemos kuriamos išskirstytų maišos lentelės pagrindu (DHT) pagrindu (Li, 2008).

Struktūrizuotose P2P sistemose kiekvienas mazgas taip pat, kaip kiekvienas išteklius, identifikuojamas identifikatoriumi, veikiantis kaip mazgas ar išteklių identifikatorius (ID). DHT apibrėžia artimiausią išteklių išsidėstymą tam tikruose mazguose, tinklo viduje.

Struktūrinės P2P topologijos užtikrina, kad mazgai žinotų kitus mazgus ir jų ID. Populiariausia P2P sistema, grindžiama struktūrine topologija yra Chord, kuri naudoja žiedo struktūros topologija (Stoica, 2002).

1.4. Sesijos inicijavimo protokolo taikymai P2P tinkluose

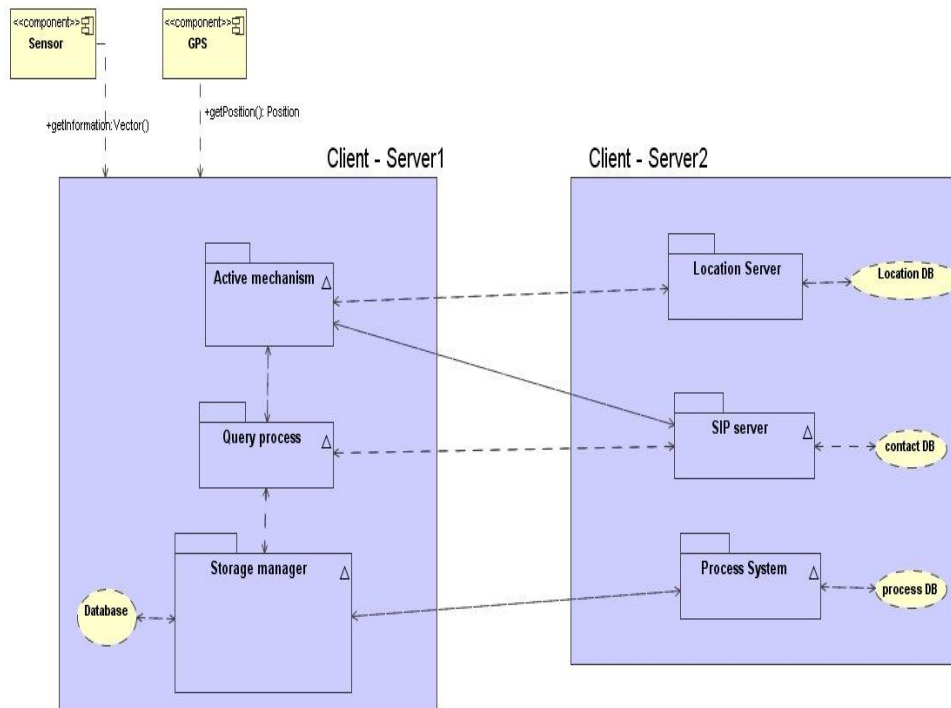
Sesijos inicijavimo protokolas SIP (angl. Session Initiation Protocol) yra dalykinės programos signalinis protokolas, kad sukurtų, pakeistų ir užbaigtų sesijas tarp vieno ar daugiau belaidžio tinklo dalyvių. SIP tapo pusiau standartu VoIP (Voice over Internet Protocol) komunikacijoje. Pagal Internet Engineering Task Force (IETF), SIP, priimtas sesijos valdymui 3GPP (Third Generation Partnership Project) projekto realizacijoje, kuria siekiama teikti naujos kartos infrastruktūrą mobilaus ryšio telefonų sistemoms (žr.1.2 pav.). SIP grindžiamas kliento-serverio infrastruktūra, kurioje vartotojo agentai yra galiniuose terminaluose, t.y. klientai, tarpiniai serveriai, tvarkantys SIP žinučių maršrutą tarp vartotojo agentų ir REGISTER serverio saugyklos, laikiančios klientų kontaktų informaciją.

Kaip rezultatas, kliento – serverio infrastruktūroje SIP tinklai charakterizuojami aukštomis administracinėmis galimybėmis (Li, 2010; Hpaslaugas, 2010; Leu, 2008) pateikta. Ripe. Todėl, pvz., Skype pradėjo integruoti to paties rango prietaisų sąveikos ("peer-to-peer") mechanizmus į VoIP komunikaciją (Islam, 2011, Shan, 2009). Dėl to, tokių tinklų infrastruktūroje buvo galima atsisakyti brangių, aukštos klasės serverių ir minimalių konfigūracijos pakeitimų kliento pusėje. Tačiau, kai Skype rėmėsi priklausančiais, neskaidriais protokolais ir telekomunikacijos kompanijos

pareikalavo tarpusavio sąveikos standartų, tai buvo mėginama integruoti P2P technologijas į SIP (P2P SIP) (Khallouf, 2008; Tseng, 2007).

Lietuvos mokslininkai taipogi prisideda prie SIP protokolo tyrinėjimų taikant šį protokolą atnaujinant balso paslaugas ir perkeldami balso ryšį į internetą (Markevičius, 2008, ir kt., Jarutis, 2011, ir kt., Pranevičius 2010, ir kt.).

SIP protokolo įvedimas, inicijuojant komunikaciją, leidžia naudotis ne tik



centralizuotomis, bet ir decentralizuotomis tinklo infrastruktūromis.

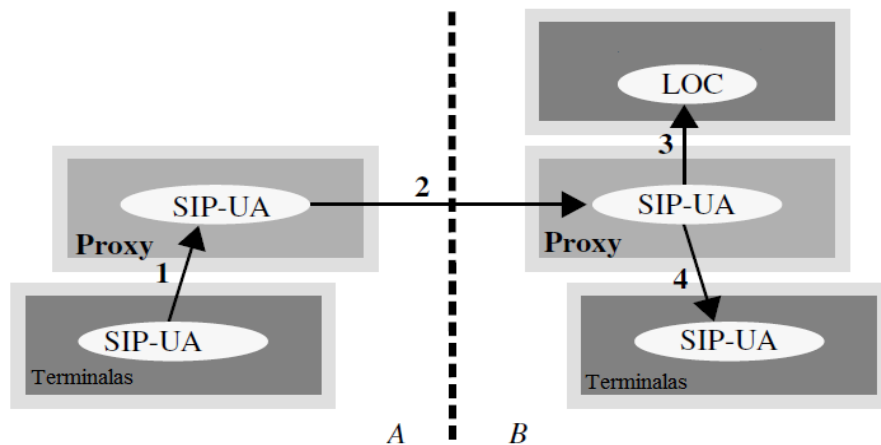
1.2 pav. Bendroji mobiliojo įrenginio ryšio sistemos architektūra (pagal 3GPP projektą)

Yra keletas metodų siekiant integruoti P2P mechanizmus į SIP: SIP protokolo išplėtimas, P2P mechanizmo integravimas į SIP žinutės perdavimo mechanizmus arba tiesioginė P2P technikos integracija į vartotojo agentą (Campi, 2009).

Tarptautinė grupė IETF sukūrė SIP protokolą ir aprašė jį RFC 3261 dokumentacijoje. SIP yra tekstinis protokolas grindžiamas sesijos valdymu. Protokolas naudojamas daugialypės terpės sesijos valdymui, pavyzdžiui inicijuoti, nustatyti ir nutraukti sesiją tarp kelių dalyvių. Dalyviai gali būti

asmenys (videokoferencinis ryšys) arba automatai (balso pašto serveriai), arba aparatai, kurie gali bendrauti.

SIP RFC nurodo tam tikrus subjektus kliento - serverio architektūroje. Vartotojo agentai (SIP-UA) siunčia ir priima SIP pranešimus, nustato, keičia ir nutraukia sesijas. Galiniai terminalai registruoja savo esamą kontaktinę informaciją (pvz. IP adresą ir portą) registratoriaus serveryje naudodami REGISTER žinutę. Tada registras kaupia šiuos duomenis vietos nustatymo paslaugoje (LOC), kuri atstovauja tam tikrą duomenų bazę dėl kontakto informacijos dalyvaujančių vartotojų agentų specifinės srities viduje.



1.3 pav. SIP srauto nukreipimas tarpiniame serveryje

Tarpiniai serveriai kontroliuoja SIP tinklo srautą (1.3 pav.). SIP sesijai inicijuoti galiniai terminalai siunčia INVITE žinutę, kurioje yra įrašytas galinio terminalo adresas SIP URI formatu (username@domainname).

Yra keli būdai sujungti P2P mechanizmus į SIP, kad pašalintų centrinių serverių reikalingumą. Tai yra standartinis SIP išlėtimas arba SIP įranginio vientisumo praplėtimas galinėse taško vietose, t.y. SIP vietos paslaugos (Mahmoudi, 2011).

SIP išplėtimas pavadintas dSIP, kuris naudoja P2P mechanizmą, palaikomą P2P praplėtimo pagrindu, veikiantį kaip registro serveris resursų lokalizacijai. Protokolo išplėtimas leidžia publikuoti ir klausinėti kontakto informacijos, kuri yra reikalinga tam, kad nukreiptų SIP žinutes be centrinio serverio (Touceda, 2011). Todėl yra svarbu įdėti SIP antraštes.

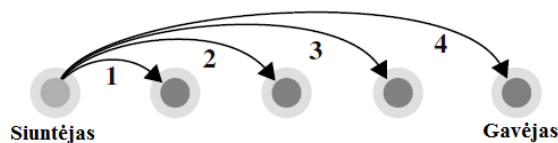
dSIP yra suprojektuotas, kad palaikytų įvairius DHT protokolus. dSIP lygiarangiai (peer) (t.y., terminalai, kurie yra suderinami su dSIP) turi aprūpinti panašų į serverio infrastruktūrą palaikantį funkcionalumą, t.y., jie turi veikti kaip registras ir tarpinis serveris tuo pačiu metu.

Apskritai P2P infrastruktūra reikalauja mechanizmų laikyti ir gauti resursų informaciją (Yi, 2010). dSIP naudoja SIP REGISTER žinutes surišti SIP resurso vardus (pvz., SIP URI) į tam tikras kontakto detales. Todėl kiekvienas lygiarangis resursas turi pasauliniu mastu unikalų identifikuoja dSIP infrastruktūroje.

Kad nusiųstų žinutę, sesijos sukūrimui, lygiarangiai įrenginiai pirmiausia turi atrasti išteklių vietą. dSIP naudoja REGISTER žinutes ištekliui atrasti. REGISTER žinutė yra be kontakto lauko įgyvendinimo, todėl lygiarangiai turi apskaičiuoti gavėjo resursą ID pagrindu šaltiniui SIP URI, parentą DHT algoritmais. dSIP apibrėžia šaltinio URI identifikuoja resursus (pvz., SIP URI) pavyzdys atrodytų, sip:bob@xyz.org; resource-ID=1257ebd371 (resurso ID vaizduoja SIP URI maišos vertę) (Lee, 2008).

Lygiarangis įrenginys kreipiasi į dinaminę maišos lentelę (DHT), kurioje yra lygiarangių įrenginių duomenys ir pasirinkdamas artimiausias žinomus įrenginius atsižvelgiant į resurso šaltinio ID, tada registracijos žinutė nusiunčiama visiems artimiausiems žinomiems lygiarangiems įrenginiams (Feng, 2011; Schmidt, 2008).

dSIP įgalina apibrėžti tris skirtingus galimus komunikacijos procesus: iteratyvų (iterative) (1.4 pav.); rekursyvų (recursive) (1.5 pav.) ir mišrų mechanizmą.



1.4 pav. Žinučių perdavimo iteratyvus mechanizmas pagal (Feng, 2011)



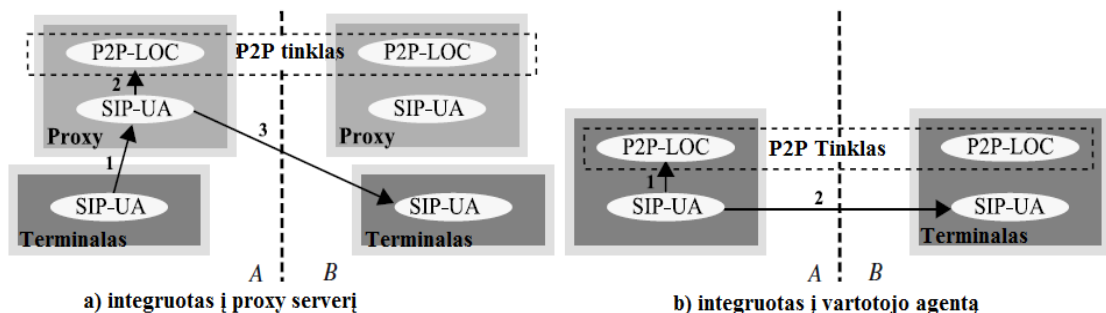
1.5 pav. Žinučių perdavimo rekursyvus mechanizmas pagal (Ferng, 2011)

Visais trimis atvejais, paskutinis atsakymas yra žinutė su kodu 200, tai reiškia sėkmingai surastą kontakto informaciją. Jei gauname žinutę, kurios kodas 404, tai reiškia nesurastą galinį terminalą. Pasikartojančiam nukreipimui, šaltinis (lygiarango siuntimas) gauna 302, peradresuoja atsakymą.

Pradinei registracijai tinkle reikia dviejų nuoseklių žingsnių: pagrindinės registracijos lygiarango integracijai į P2P tinklą ir SIP kontakto informacijos, kad būtų pasiekiami kitiems SIP terminalams. Pirmiausia įrenginiai lygiarangiame tinkle siunčia registracijos žinutę su identifikatoriumi (Node ID).

1.5. SIP lokacijos paslaugos išplėtimas

SIP lokacijos paslaugos praplėtimas SIP lokacijos paslauga (P2P-LOC), skirtingai nuo standartinių SIP lokacijos paslaugų, kaupia šalia esančių kontaktų informaciją P2P tinkle. Lokacijos paslaugos sąsaja aprūpina priemones registracijai, nustatant SIP-URI kontaktinę informaciją. P2P-LOC gali būti integruojama į SIP tarpinį serverį (P2P-proxy) arba tiesiogiai į vartotojo agentą (P2P-UA). Mišrus P2P-Proxy ir P2P-UAS taip pat yra galimas (1.6 pav.). SIP lokacijos paslaugos praplėtimas reikalingas kada nėra centralizuotos sistemos to pasekoje kiekvienas įrenginys turi žinoti visų kitų įrenginių kontaktų informaciją.



1.6 pav. SIP lokacijos serviso išplėtimas pagal (Shim, 2011)

SIPpeer, tai sukurtas P2P SIP adapteris, leidžiantis dalyvauti P2P tinkle, be pakeitimų vartotojo agente įgyvendinamas P2P-proxy (Shim, 2011).

1.6. Judančių objektų SIP komunikacijos galimybės 3GPP tinkle

Pagal Europos bendruomenių komisijos „Commission of the European Communities“ pasirinktą trečios kartos tinklo 3GPP pagrindinę techninę specifikaciją yra apibrėžiami siuntimo iš transporto priemonės mobiliam operatoriui protokolai (Michalski, 2009). Tai atvirasis standartas. Didėjant telekomunikacijos įvairovei, atsiranda naujos kartos duomenų perdavimo galimybės. Žiūrint į specifikaciją „3GPP TS 24.228 Release 5 "Signalling flows for the IP multimedia call control based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP)“ informacijos perdavimas grindžiamas IP funkcionalumu (Handley, 2006; Garcia, 2005).

Norint apjungti kuo daugiau įrenginių, reiktų turėti vieningą protokolą, apjungiant įrenginius tarpusavyje. Remdamiesi išskirstytomis sistemomis transporto vartotojai gali efektyviau atlikti sudėtingas užduotis, bendradarbiauti ir koordinuoti savo veiksmus. Transporto vartotojų dalinimasis resursais su kitais transporto vartotojais, turi turėti gebėjimą nuskaityti parametrinius duomenis iš nutolusių sistemų. Organizuojant sąveikumą, tenka ieškoti būdų juos integruoti ir jų galimybes sujungti į vieną bendrą protokolą. Šiame darbe sprendžiama SIP galimybių integracijos problema, suteikiant sąveikumą ir komunikavimą tarp skirtingų transporto priemonių (įrenginių). Tiriant belaidžių įrenginių komunikavimo protokolo sistemos aprašymą, nagrinėjamos sesijos inicijavimo protokolo (SIP) praplėtimo galimybės (Hao, 2008; Hilt, 2008). Sprendžiamos problemos, susijusios su ryšio palaikymu tarp skirtingų įrenginių, pasinaudojus vieningu protokolu, šiuo atveju – sesijos inicijavimo protokolu (Asgharian, 2011). Sensorinius ar nutolusius įrenginius integruojant į bendrą tinklą, grindžiamą SIP protokolu, numatoma, kad tokia technologija pakeis informacinių sistemų ir telekomunikacijų rinką (Fokum, 2010).

Šios srities ekspertai prognozuoja technologijos proveržį artimiausiais metais. Proveržis turėtų būti paskatintas mažėjančiomis komponentų kainomis ir platesniu technologijų pasiekiamumu. Sesijos inicijavimo protokolas SIP (angl. Session Initiation Protocol) yra dalykinės programos signalinis protokolas, kuris sukuria, pakeičia ir užbaigia sesijas tarp vieno ar daugiau belaidžio tinklo dalyvių. SIP tapo pusiau standartu VoIP (Voice over Internet Protocol) komunikacijoje (Andreasen, 2006; Adeyeye, 2009; Barnawi, 2012). Pagal IETF bendruomenę, SIP priimtas sesijos valdymui, kuriuo siekiama teikti naujos kartos 3GPP (Third Generation Partnership Project) infrastruktūrą mobilaus ryšio telefonų sistemoms. SIP grindžiama kliento-serverio infrastruktūra, kurioje vartotojo agentai yra galiniuose terminaluose t.y. klientai, tarpiniai serveriai tvarkantys SIP žinučių maršrutą tarp vartotojo agentų ir REGISTER serverio saugyklos, laikančios klientų kontaktų informaciją (Subramanian, 2010).

Didėjant telefonijos vartotojų poreikiui naudotis internetu, tuo pačiu metu atsirado poreikis telefonų komunikacijose įtraukti ir internetą. Mobilųjų įrenginių operatoriai, turint greitą interneto prieigą, negali daugiau sau leisti teikti tik balso paslaugas, kadangi tai tapo įprastinėmis paslaugomis. Operatoriai turi teikti patrauklias paslaugas, kurios gali būti pateikiamos su baziniu prieigos teikimu. Mobilųjų įrenginių paslaugų teikėjai negali pasiūlyti multimedijos komponentų (teksto, nuotraukų, video, garso) sujungimo vieno skambučio metu (Gurbani, 2011). Dvipusis garso ir vaizdo grupinis telefoninis pokalbis negalimas (t.y. konferencija negalima) (Barbosa, 2011). Tačiau IMS išsprendė šį apribojimą. IMS plečiamumas leidžia duomenų ir telekomunikacijos paslaugų integraciją (Chu, 2012, Homayouni, 2009). Tam, kad galėtume naudotis interneto teikiamais privalumais, atlikus minimalias modifikacijas esamai architektūrai, buvo sukurta IMS architektūra (Yeganeh, 2011), kurios pagrindiniai privalumai buvo tokie:

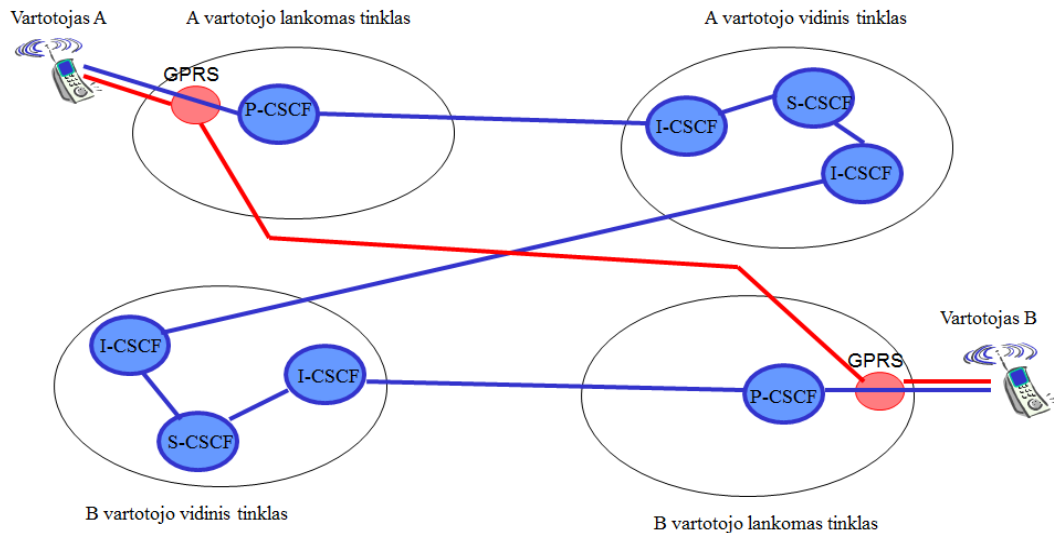
- Integracija su plačiai paplitusiomis esamomis architektūromis, kas leidžia IMS sujungti GSM, VoIP, standartinius bei kitus telefonijos tinklus.

- Duomenų perdavimas interneto protokolu (IP) ir tuo pačiu metu galimybę teikti kitas internetines paslaugas telefonu tinkle leidžia SIP (pvz.: konferencija, video telefonija, kitos paslaugos).

IMS – tai IP multimedijos posistemis. IMS yra IP paremtas tinklas, kuris jungiasi prie įvairių prieigos taškų, suteikiantis vieną paslaugą belaidžio, linijinio ir kabelinio tinklo vartotojams. Iš pradžių IMS modelis buvo sukurtas išplėsti GSM mobiliųjų tinklų galimybes, tačiau vėliau buvo pridėtos galimybės palaikyti kitus tinklus ne tik GPRS (pavyzdžiui, belaidis tinklas WLAN, CDMA2000 bei laidinis tinklas. IMS (IP Multimedia System). IMS – nauja posistemė, t.y. nauja mobiliojo ryšio infrastruktūra, kuri leidžia duomenų, balso ir mobiliojo ryšio technologijų konvergenciją per IP infrastruktūrą (Oredope, 2011).

IMS suteikia lanksčią IP multimedijos valdymo ir sesijų kontrolės platformą, kurią operatoriai gali aprėpti esama tinklo infrastruktūra. Skambučiams valdyti IMS naudoja sesijos inicijavimo protokolą, kurio metu skambutis (sesija) yra perduodama tarp skirtingų skambučio valdymo elementų. IMS yra tinklo funkcinė architektūra, kuri palengvina multimedijos paslaugų kūrimą bei diegimą ir palaiko sąveikas (Chu, 2012). IMS architektūra konkrečiai padeda įgalinti ir pagerinti realaus laiko daugialypės terpės mobiliąs paslaugas, pavyzdžiui, balso paslaugas, vaizdo telefoniją, pranešimus, konferencijas ir kitas paslaugas. IMS pats savaime nėra technologija, o yra architektūra, paremta interneto standartais, kuriais šiuo metu pristatomos paslaugos tinkle. IMS architektūra naudojama įgalinti sesijas multimedijos pagrindu tarp keleto ar daugiau įrenginių. IMS centrinis tinklas apibrėžiamas kaip sluoksninis tinklas, susidedantis iš multimedijos transportavimo sluoksnio (garsas, duomenys, video ir kt.), skambučių kontrolės sluoksnio (SIP kontrolė, perdavimo kontrolė ir kt.) ir programų sluoksnio (suteikiantis papildomą funkcionalumą IMS centriniam tinklui). Pagrindinis IMS centrinio tinklo privalumas yra bendras programų sluoksnio naudojimas. Bendri skambučio kontrolės ir transporto sluoksniai vartotojams suteikia įvairių paslaugų tarp skirtingo prisijungimo tinklo. Visos sesijos signalizacijos yra perduodamos per vidinius tinklus kiekvienam

abonentui. Nutolę mobiliųjų įrenginių vartotojai gali bendrauti vienas su kitu komunikavimo platformoje kaip pateikta 1.7 pav.



1.7 pav. Mobilųjų įrenginių komunikacijos schema (Chu, 2012 ir Oredope, 2011)

Duomenų perdavimo per GPRS paslaugų teikimui naudojama sąsaja - raudona linija (žr 1.7 pav). Mėlyna linija naudojama SIP ir SDP atvaizduoti.

Bendrojo paketinio radijo ryšio paslauga (General Packet Radio Service – GPRS) – tai tarpinė grandis tarp GSM ir 3G korinių tinklų. GPRS įgalina spartesnę duomenų perdavimą (nuo 9,6 iki 115 kilobitų) per GSM tinklą. Ja naudodamiesi vartotojai gali skambinti ir perduoti duomenis tuo pat metu. Pavyzdžiui, jei turite GPRS mobilųjį telefoną, jūs galite skambinti ir gauti elektroninius laiškus tuo pat metu. Pagrindinis GPRS privalumas – radijo tinklo ištekliai rezervuojami tik tuomet, kai perduodami duomenys, todėl galima mažiau pasikliauti standartiniais komutuojamo ryšio tinklo elementais.

IMS CSCF - Call Session Control Server (skambinimo sesijos kontroliuojantis serveris), yra dalis architektūros, kuris suteikia galinių įrenginių registraciją. CSCF naudojama apdoroti SIP signalinius paketus IMS architektūroje. Jis taip pat numato SIP žinutės maršrutizavimą. IMS CSCF gali būti suskirstytas šitaip:

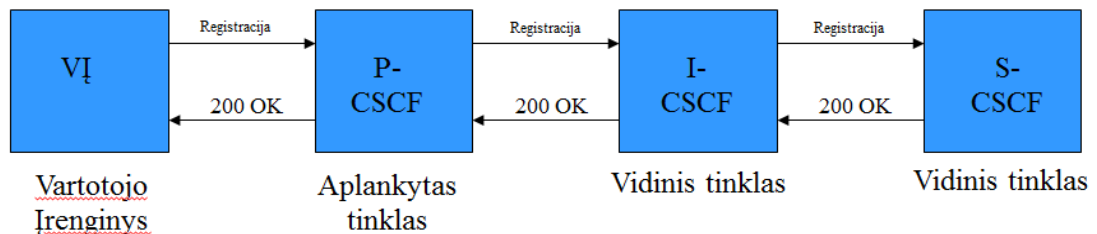
P-CSCF yra dalis, kur įrenginiai patenka į IMS architektūrą. P-CSCF yra kontakto pirminis taškas vartotojo įrenginiui persiunčiant SIP žinutes į S-CSCF.

Taip pat suteikia įrenginio valdymo tarptinklinį saugumą. P-CSCF yra vartotojas, nukreipiantis į tinklą. Šiuo atžvilgiu visa SIP signalizacija pareina per P-CSCF vidiniame namų tinkle ar kitos šalies tinkle. Kai tik vartotojas registruojasi IMS tinkle, registracijos signalizacija praeina per P-CSCF.

S-CSCF (server-CSCF) – elementas bendroje IMS SCSF architektūroje yra sesijos valdymas galiniams įrenginiams ir išlaiko sesijos būseną.

I-CSCF (Interrogating (apklausiantysis) CSCF) – yra sesijos valdymo subjektas galiniams įrenginiams, kurie palaiko sesijos būsenas. I-CSCF naudojamas perduodant pradinę SIP užklausą į S-CSCF, kai inicijatorius nežino, kuris S-CSCF turi gauti užklausą. Pagrindinis uždavinys – rasti S-CSCF registracijos metu.

HSS (Home Subscriber Server) – svarbus elementas IMS architektūroje, kurioje laikomi abonentų duomenys. Kai abonentai registruojasi IMS tinkle, abonentų duomenys gaunami iš HSS per S-CSCF, kurie buvo priskirti abonentui. P-CSCF, I-CSCF ir S-CSCF egzistuoja kiekvienam vartotojui ir yra nustatyta registracijos metu kaip nurodyta (žr. 1.8 pav.).



1.8 pav. (šaltinis: Chu, 2012 ; Oredope, 2011) SIP registracijos žinutės perdavimo struktūra IMS architektūroje

Registracija identifikuoja S-CSCF. Visos naujos išeinamos sesijos, identifikuojamos pagal FROM/To/CALL-ID antraštes, turi būti nukreiptos pagal P-CSCF į S-CSCF. Visos naujos įeinamos sesijos, identifikuojamos pagal FROM/To/CALL-ID antraštes, turi būti nukreiptos S-CSCF į P-CSCF. Siunčiant INVITE užklausą į nutolusį vidinį tinklą, kur I-CSCF atlieka būtinas paieškas surasti nutolusio vartotojo S-CSCF. Vartotojas visada nukreipiamas į tą patį P-CSCF. Valdymo sluoksnio integracija per vieningą prieigą prie fiksuotojo ir

judriojo ryšio pagrindinio tinklo, suteikti vartotojams vieningą skambučių ir sesijos kontrolę.

1.7. Pirmojo skyriaus išvados

1. Atlikta sesijos inicijavimo protokolo architektūrinių sprendimų analizė ir nustatyta, kad sesijos užmezgimas, palaikymas ir nutraukimas galimas lygiaranguose (Peer-to-Peer) ir 3G kartos tinkluose naudojantis vieningu SIP protokolu, kas leido naudotis skirtingomis tinklo topologijomis nenaudojant daugybės kitų protokolų. Palygintos skirtingų tinklų topologijos duomenų keitimuisi ir padaryta išvada, kad šiais tinklais įrenginiai gali užmegzti sesijas SIP protokolu.
2. Apžvelgtos judančių objektų stebėsenos galimybės, kurios leidžia taikyti mobiliąsias technologijas. Aptariamos kylančios problemos, galinčios įtakoti operatyvių sprendimų priėmimus ir nagrinėjamos jų sąveikavimo aprašymo galimybės.
3. Išplėtos sesijos inicijavimo protokolo funkcinės savybės leido nustatyti judančio objekto geografinę poziciją, pasinaudojus įrenginio fiksaciją per SIP adreso žymę, kas leido nustatyti poziciją, ten kur neturi galimybių veikti GPS.

2. DUOMENŲ KEITIMASIS TARP NUTOLUSIŲ HETEROGENINIŲ SISTEMŲ TAIKANT SESIJOS INICIJAVIMO PROTOKOLO GALIMYBES

Šiame skyriuje pateikiamas SIP veikimas, apžvelgtos galimybės šio protokolo integracijai, kad galėtų komunikuoti skirtingi nutolusieji įrenginiai. Pateikiamas SIP daugiasluoksnė architektūra ir aprašoma šio protokolo keturi sluoksniai, kurių kiekvienas vykdo atitinkamus funkcijų rinkinius. Apžvelgiama keturių skirtingų nutolusių heterogeninių įrenginių tipų integravimo ir veikimo ypatybės bei galimybės sujungiant į komunikavimo tinklą.

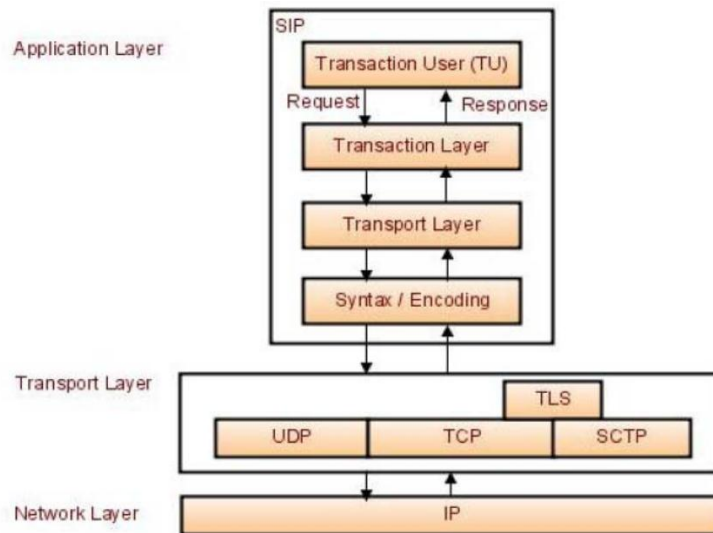
2.1. SIP veikimo aprašymas

Vieni sėkmingiausių projektų naudojantys SIP protokolą yra XLITE (Xlite, 2013), EKIGA (Ekiga, 2013), SKYPE (Skype 2013). Skype šiuo metu yra uždaras protokolas, t.y., duomenys šifruojami.

SIP protokolą suprojektavo IETF Multi-Party Multimedia Session Control Working Group, žinoma kaip MMUSIC. Protokolas kaip standartas buvo paskelbtas RFC 2543, 1999 metais ir 2001 metais buvo pataisytas ir paskelbta galutinė versija. SIP (Session Initiation Protocol) – standartas, kuris palengvina ryšio formavimą, modifikavimą ir vykdymą tarp dviejų ar daugiau dalyvių, patvirtintas Internetinės inžinierių grupės (IETF). Žemiausias yra fizinis sluoksnis, kuris gali būti vietinis tinklas (LAN) arba telefono linijos 56K modemas, veikiantis kaip Point-to-Point protokolas (PPP) arba digital subscriber line (DSL), veikiantis asinchroniniu režimu. Šie sluoksniai atlieka tokias funkcijas kaip duomenų srautų mainai, sinchronizacija ir fizinės sąsajos specifikacija.

Interneto sluoksnis – interneto protokolas (IP), naudojamas nukreipti paketus tinkle, naudojantis paskirtu IP adresu. IP adresas – kompiuterio identifikatorius IP tinkluose. Tai unikalus skaičius, naudojamas vienareikšmei duomenų paketo siuntėjo ir gavėjo identifikacijai ir skiriamas žmogaus ar organizacijos, administruojančios duotąjį IP tinklą. Internete IP adresus skiria tam tikrus jų režius turinčios organizacijos. Šiuo metu naudojama sistema skaitiniams

adresams naudoja 32 bitus, todėl teoriškai iš viso galima panaudoti virš keturių milijardų adresų. Praktiškai šių adresų yra žymiai mažiau, nes nemaža dalis naudojama specialioms reikmėms. Kadangi laisvų 4-os versijos adresų mažėja, raginama pereiti prie 6-os IP versijos.



2.0 pav. Sesijos inicijavimo protokolo taikomojo sluoksnio integracija su transporto ir tinklo sluoksniu (pagal IETF MMUSIC rekomendacijas)

SIP transporto sluoksnis (angl. Transport layer) – tiekia skaidrų duomenų perdavimą tarp tinklo vartotojų su norimomis patikimumo garantijomis. Pvz.: TCP, UDP. Riba tarp sesijos ir transporto lygių gali būti traktuojama kaip riba tarp taikymo lygio protokolų ir žemutinio lygio protokolų. Transporto lygis užtikrina patikimą duomenų siuntimą (žr. 2.0 pav.). Šiame lygyje atliekamas aukštesnių lygių taikomųjų programų segmentavimas ir išskyrimas į vieną bendrą transporto lygio duomenų srautą. Taip pat sukuriama loginis ryšys tarp dviejų siuntimo taškų.

SIP dalykinių programų sluoksnis (angl. Application layer) – aukščiausias lygis, apibrėžiantis tinklo teikiamas paslaugas vartotojo programoms. Šis lygis apibrėžia taikomas programas, kurioms reikia siųsti informaciją tinklu, t. y. normaliam darbui reikalingi tinklo resursai. Jei taikomajai programai nereikalingas duomenų siuntimas, tai tinklo resursai jai nereikalingi ir ji nepriklauso šiam lygiui. Skiriamos tokios taikomųjų programų rūšys: tinklo

taikomosios programos (network applications) ir tarptinklinės taikomosios programos (internet work applications).

SIP sesijos sluoksnis (angl. Session layer) – aprašo duomenų apsikeitimo tarp galinių sistemų taisykles vieno sujungimo ribose. Sesijos lygyje užmezgamos, valdomos ir nutraukiamos tarp taikomųjų programų. Šis lygis sinchronizuoja dialogą tarp atvaizdavimo lygių ir valdo jų duomenų mainus papildomai, be sesijų valdymo (užmezgimo, nutraukimo).

SIP ryšio sluoksnis (angl. Data link layer) – aprašo ryšį tarp gretimų (tiesiogiai bendraujančių) tinklo komponentų, pavyzdžiui, kadru struktūrą. Pagrindinė funkcija – atskirti bitų sekas, jų pradžią, pabaigą. Ryšio lygis dar vadinamas duomenų ryšio lygiu, užtikrina duomenų transportą per fizinį ryšį. Ryšio lygis susietas su fiziniu adresavimu, tinklo topologija, klaidų notifikacija (pranešimais apie klaidas), paketų sutvarkymu ir duomenų srauto kontrole. Čia duomenys suskirstomi į freimus, aptinkamos ir ištaisomos perdavimo klaidos, nustatomi fiziniai adresai. SIP protokolas bendrauja su dviem internetiniais protokolais – TCP ir UDP. TCP – tai vienas iš pagrindinių protokolų, esančių Internetinių protokolų rinkinyje (angl. Internet protocol suite). TCP yra tarpinis lygis tarp IP ir aplikacijos bei priklauso transportavimo lygmeniui. Naudodamos šį protokolą, aplikacijos gali sukurti sesijas tarp kelių tinklo taškų ir dalintis duomenimis. Priešingai nei UDP, šis protokolas užtikrina patikimą duomenų perdavimą tarp dviejų tinklo taškų. UDP tai TCP/IP naudojamas perdavimo protokolas. Šis protokolas yra alternatyva TCP protokolui. Skirtingai nei TCP, UDP nėra patikimas, neatlieka duomenų tėkmės kontrolės (angl. flow-control) ir neturi klaidų atitaisymo mechanizmų, UDP naudojamas tik persiųsti duomenis.

Vartotojai turi adresus (pvz., elektroninio pašto adresas), kurie leidžia identifikuoti ir nustatyti paskirties vietą. SIP galime išskaidyti į tris atskirus protokolus:

1. Sesijos inicijavimo protokolas (RFC2543) (Siunčia žinutes).
2. Sesijos aprašymo protokolas (RFC2327) (Session Description Protocol (SDP))

3. Informacijos perdavimo protokolai.

SIP susideda iš penkių komponentų:

Skambinantysis (User Agent Client (UAC));

Atsakantysis (User Agent Server UAS);

Tarpinis serveris (proxy server);

Nukreipiantysis serveris (redirect server);

Registratorius serveris (Registrar server).

Skambinantysis yra vienas iš dviejų dalyvių, kuris pradeda kontaktą su kitos pusės dalyviu (UAS). Skambinančioji pusė gali pasiųsti šešias SIP užklausas: Kvietimas (INVITE), Patvirtinimas (ACK), Pasirinkimas (Options), Atlaisvinimas arba Užbaigimas (Bye), Atšaukimas (Cancel) ir registravimas (Register). Kai SIP sesija pradama, UAC būtina informacija apie UAS buvimo vietą ir IP adresą. Ši informacija gali būti dinamiška ir todėl gali iškilti problemų dėl sujungimo.

Atsakantysis serveris priima signalus iš UAC serverio ir atsako į juos.

Tarpinis serveris yra tarpininkas, prižiūrintis siunčiamas užklausas ir atsakymus į jas. Tarpinis serveris gali būti vidinis (kai kontaktuojama organizacijos viduje, žinutės yra siunčiamos per tą patį serverį) ir gali būti keli išoriniai serveriai, kai komunikuojama tarp kelių organizacijų, tuomet serveriai komunikuoja tarpusavyje.

Nukreipiantysis serveris sudaro sąlygas nukreipimui, kuris suteikia galimybę vartotojams laikinai keisti geografinę padėtį ir išlaikyti kontaktą, išlaikant tą patį SIP identiškumą.

Registratoriaus serveris suteikia galimybę vartotojams persiųsti adresus iš kontaktuojančių įrenginių. SIP klientas siunčia registruotą užklausą apie adreso pakeitimą į registrų serverį, kuris priima užklausą ir įrašo vartotojo naująjį adresą. SIP vartotojui bendrauti su registrų serveriu yra du būdai. Pirmas būdas – tiesiogiai, panaudojant informaciją, kuri yra konfigūruojama kliento. Antras būdas – netiesioginis, kai įvairūs klientų adresai kontaktuoja su serveriu.

SIP pranešimais operuojantis protokolas, kuris naudojasi užklausomis ir atsakais ryšiui tarp įvairių tinklo komponentų užmegzti (Rosenberg, 2002).

Vartotojai SIP tinkle yra identifikuojami pagal unikalų SIP adresą. SIP adresas panašus į elektroninio pašto adresą: vartotojoID@sietuvas. Vartotojai registruojasi registravimo serveryje su jiems priskirtais SIP adresais (Rusinovic, 2009). Kadangi duomenų paketai juda nepriklausomai vienas nuo kito tiek maršruto, tiek laiko prasme, todėl informaciją gaunantis kompiuteris iš siunčiančio kompiuterio turi užklausti informacijos apie trūkstamus paketus. Gaunant paketus reikalaujama reikiama tvarka sutvarkyti gautus paketus, atkoduoti juose esančius duomenis bei paversti atitinkama informacija. Tokiu būdu gali būti perduodama bet kokio pobūdžio informacija, kuri yra skaitmenizuota (Sarikaya, 2008, Cheng, 2008).

Kai vartotojas inicijuoja skambutį, SIP užklausa pasiunčiama SIP (įgaliotajam arba peradresavimo serveriui) (Chi, 2008). Užklausoje yra skambinančiojo ir gavėjo adresas.

Įgaliotasis serveris suranda maršrutą ir persiunčia užklausą gavėjui (Sun, 2007). Gavėjas atsako įgaliotajam serveriui, kuris persiunčia atsaką skambinančiajam. Įgaliotasis serveris persiunčia abiejų šalių užklausas, sudaroma sesija tarp skambinančio ir gavėjo. Šiuo etapu skambinantysis ir gavėjas bendrauja per RTP protokolą.

Pagrindiniai SIP žinutę formuojantys laukai yra šie:

- INVITE žinutės laukai:
- INVITE sip:B_Vartotojas@serveris.org SIP/2.0
- Via: SIP/2.0/UDP lab.serveris.org:5060
- To: B_Vartotojas<sip:B_Vartotojas@radio.org>
- From: A_Vartotojas<sip:A_Vartotojas@serveris.org>
- Call-ID: 123456789@lab.serveris.org
- CSeq: 1 INVITE
- Subject: Tęstinis pranešimas...
- Contact: sip:A_Vartotojas@serveris.org
- Content-Type: application/sdp
- Content-Length: 158

Aprašoma metodika ir programinės įrangos sąsajos funkcijos, kuriomis galima pasiekti SIP protokolo funkcionalumą sesijai sukurti, palaikyti ir nutraukti, kuriant vieningą komunikacijos teikimą skirtingose aplinkose. Mūsų siūlymai įgalina komunikaciją tarp heterogeninių išskirstytų objektų. Pirmiausia klasifikuosime objekto ypatybes, galimybes ir tinklo sujungimo pobūdį. Atsižvelgiant į šią klasifikaciją ir programinį palaikymą pasirenkama SIP komunikacijos magistralė. Patikrinta skirtingų objektų tarpusavio komunikacija, įskaitant ZigBee, X10, telefonai, SIP telefonai, internetinės kameros ir kiti prietaisai SIP komunikacijos magistralėje (Kim, 2007).

Tai įrenginiai, plačiai naudojami tinklo aplinkoje pradedant namo valdymu, baigiant sveikatos priežiūrai naudojamais įrenginiais.

Šie objektai daugeliu atžvilgių yra skirtingi:

- skirtingi įrenginiai arba programinė įranga;
- skirtingi tinklo sluoksniai (IP, ZigBee, X10);
- sąveikauja skirtingais komunikavimo būdais;
- keičiasi skirtingais duomenimis (tam tikrų sensorių parametrais ar garsiniais pranešimais).

Šie įrenginiai tinklo aplinkoje yra dinamiški, objektai laiko momentu keičia savo būsenas (telefonas įjungiamas / išjungiamas).

Sprendžiama belaidžių įrenginių komunikacijos pritaikymo problema taikant vieningą SIP protokolą heterogeninėse ir dinamiškose aplinkose (Zhang, 2009). Šis standartas – internetinės telefonijos pagrindas, suteikiantis galimybę keistis duomenimis tarp įrenginių. Skirtingomis komunikacijos rūšimis galima sėkmingai apsikeisti tam tikrais ryšių formatais, tokiais kaip multimedijos sesijos ar tam tikros žinutės.

Realizuojant komunikacijos sprendimus siūlome taikyti Java programavimo kalbą. Jos priemonėmis plėtoti šias paslaugas. Tam tikrais atvejais galinius įrenginius reikės programuoti C kalba.

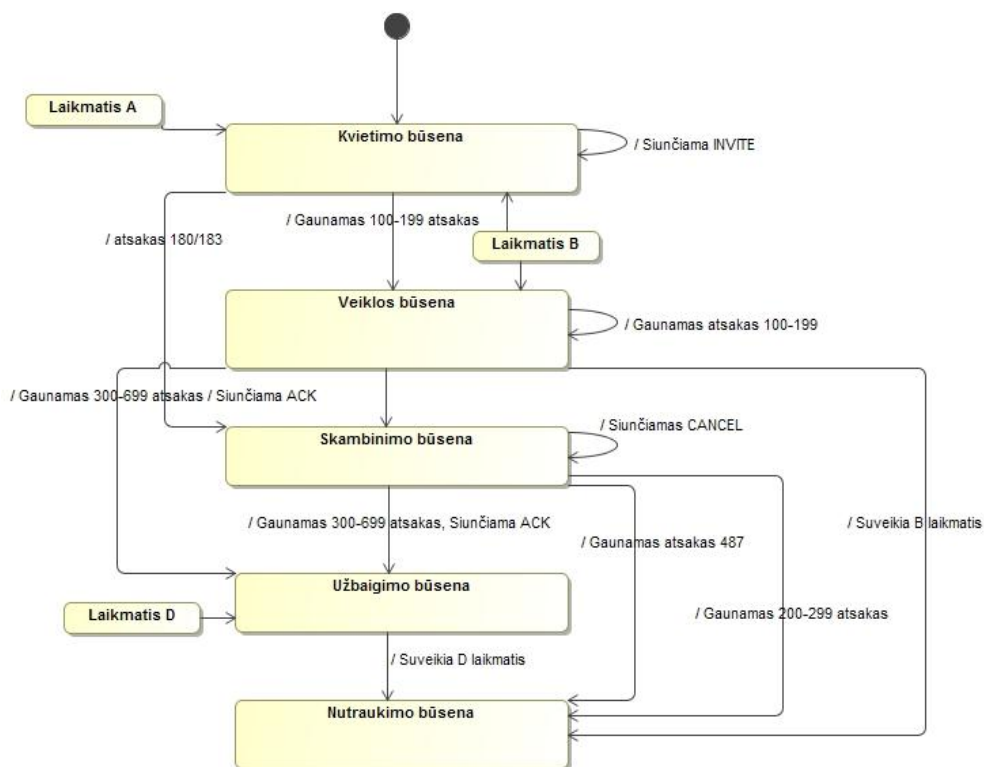
2.2. Sesijos inicijavimo protokolo sluoksniai ir būsenos

Sesijos inicijavimo protokolas suskirstytas į sluoksnius, kurių kiekvienas vykdo atitinkamus funkcijų rinkinius. SIP protokolas naudoja tradicinių tinklo (IP) sluoksnių, transportavimo sluoksnių (UDP, TCP, TLS) ir taikomųjų sluoksnių struktūrą (Barnawi, 2012). Taikomasis sluoksnis sudarytas iš:

- komandų sintaksės ir kodavimo;
- transportavimo sluoksnio;
- transakcijos sluoksnio ir kliento transakcijos sluoksnio.

Sintaksės ir kodavimo sluoksnis apibrėžia SIP žinutės struktūrą. Transportavimo sluoksnis išsiunčia arba priima SIP žinutes iš pagrindinės transportavimo aplinkos. SIP transportavimo sluoksnis yra transakcijos sluoksnis, kuriame kiekviena transakcija susideda iš kliento transakcijos siunčiant užklausas ir serverio transakcijos atsakant į užklausas (Khoury, 2007, Wu, 2010). Aukščiausias sluoksnis yra transakcijos vartotojas, kuris kuria ir naikina SIP transakcijas ir yra naudojamas teikti paslaugas transakcijos sluoksnyje (Zou, 2007).

Svarbiausias SIP sluoksnis yra transakcijos sluoksnis, atsakingas už užklausų ir atsakų žinučių suderinimą, žinučių retransliavimą, esant nepatikimai transportavimo terpei (Wu, 2007). SIP žinutės yra siunčiamos užklausomis iš kliento į serverį ir atsakomos iš serverio klientui. Kiekviena užklausų žinutė vykdoma metodais, tokiais kaip INVITE arba ACK, iškviečiant konkrečią operaciją serveryje. Kiekvienas atsakas turi statusų kodus, kuriais galima identifikuoti SIP užklausų priėmimą, atmetimą ar nukreipimą. Pirminė žinutė perduodama SIP transakcijai visada yra INVITE transakcijos užklausų žinutė arba kitos transakcijos (jei tai ne INVITE ar ACK užklausų žinutės). INVITE transakcija naudojama iškviešti sesijas, o kitos ne INVITE transakcijos naudojamos išardyti ar išlaikyti sesijas tam tikrą laiką. Kliento ir serverio INVITE transakcijos metodo apibrėžtys yra RFC 3261 specifikacijoje, aprašant per būsenų mašinas (žr. 2.1 pav.). Transakcijos klientas inicijuoja sesiją kliento pusėje, sukurdamas INVITE metodą kliento transakcijoje.



2.1 pav. Sesijos inicijavimo protokolo kliento pusės būsenų mašina (sudaryta autoriaus pagal IETF MMUSIC rekomendacijas)

Klientinės pusės transakcija (pav. 2.1) turi penkias būsenas:

1. Kvietimo (calling);
2. Veiklos (proceeding);
3. Skambinimo (ringing);
4. Užbaigimo (completed);
5. Nutraukimo (terminated).

1 lentelė.

Atsakų reikšmės į užklausas

Atsakas	Funkcija
1xx	Laikina, gauta užklausa, bet dar neatsakyta
2xx	Sėkminga, užklausa gauta ir priimta
3xx	Nukreipimas, reikalinga papildomų veiksmų
4xx	Bloga sintaksė užklausoje
5xx	Serveriui nepavyko atsakyti į užklausą

6xx	Nėra serverio galinčio atsakyti į užklausą
xx reiškia numerius nuo 00 iki 99	

Transakcija sukuriama esant skambinimo būsenai. INVITE užklausa gauta iš transakcijos vartotojo, perduodama SIP transporto sluoksniui, persiunčiant į serverio pusę ir tuo metu inicijuojamas laikmatis B. Laikmatis A yra sužadinamas tik tada, kai transportavimo terpė yra nepatikima. Kai transakcija yra skambinimo būsenoje, gali atsirasti vienas iš penkių įvykių:

1. Laikmatis A sužadinamas. Transakcija atstato laikmatį ir retransliuojama INVITE užklausa;
2. Laikmatis B sužadinamas ir transakcija įeina į pabaigos būseną;
3. Transportavimo klaida yra pranešama SIP transportavimo sluoksniu, kai bandoma siųsti INVITE užklausą tinklu. Transakcija informuoja transakcijos vartotoją apie klaidą ir pereina į nutraukimo (terminated) būseną. Kai laikinasis 1xx transakcijos atsakas (2 lentelė) gaunamas, keičiama į veiklos būseną tolimesniu atsaku.
4. Sėkmingai gautas 2xx atsakas nurodo, kad serveris priėmė INVITE užklausą. Transakcija informuoja atsaką, transakcijos vartotoją ir pereina į nutraukimo (terminated) būseną.
5. Galutinis nesėkmingas atsakas, kurio statusų intervalas 300-699, gaunamas, kai serveris nepriėmė INVITE užklauskos. Transakcija persiunčia atsaką transakcijos vartotojui, sukurdama ACK užklausą ir perduodama jį SIP transporto sluoksniui, tada būseną tampa sėkmingai įvykdyta.

Kai kliento transakcija yra veiklos būsenoje, galimi veiksmai:

- gaunamas laikinas 1xx atsakas ir liekama tyrimo būsenoje;
- gaunamas sėkmingai įvykdytas atsakas ir pereinama į nutraukimo būseną;
- gaunama sėkmingai įvykdyta būseną po ACK sukūrimo ir išsiuntimo;
- sėkmingai įvykdytos būsenos tikslas yra įsisavinti 300-699 intervalo atsakus ir retransliuoti serveriui (kai terpė nepatikima);

- kai laikmatis D inicijuojamas ir nustoja galioti, transakcija tampa nutraukimo būsenoje;
- jei 300-699 intervale atsakas yra gaunamas prieš laikmačio D pabaigą, transakcija sukuriama, siunčiamas ACK ir būseną nesikeičia;
- jei įvyksta transportavimo klaida, tada transportavimo sluoksnis siunčia ACK, informuojamas transakcijos vartotojas, transakcijos būseną tampa nutraukiama;
- kai kliento transakcija inicijuoja nutraukimo būseną, tada transakcija yra sunaikinama nedelsiant;
- kai transakcijos vartotojas gauna INVITE užklausą serverio pusėje, sukuriama serverio transakcija, išsiunčiamas trying (100) atsakymas. Jei transakcijos vartotojas nesukuria atsakymo per 200 milisekundę (ms), transakcija lieka veiklos būsenoje. Kitos trys transakcijos būsenos yra: sėkmingai įvykdytas, patvirtinimas ir nutraukimas.

Serverio transakcija turi 4 būsenas, kurios yra:

1. Veiklos (proceeding);
2. Užbaigimo (completed);
3. Patvirtinimo (confirmed);
4. Nutraukimo (terminated).

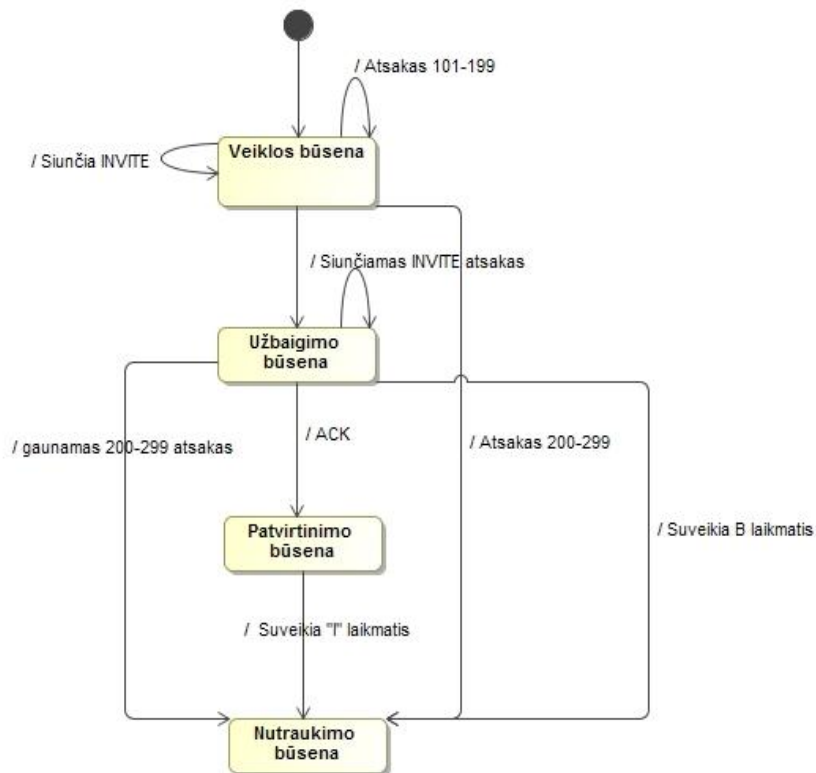
Kai serverio transakcijos yra veiklos būsenoje galimi šie veiksmai:

- perduoti bet kuriuos intervalo 101-199 laikinuosius atsakymus, sugeneruotus transakcijos vartotojo, SIP transporto sluoksniui ir būseną nesikeičia. Gaunama INVITE užklausa, retransliuojama kliento transakcija, kai serverio transakcija retransliuoja laikiną atsakymą, gautą iš transakcijos vartotojo, ir liekama tyrimo būsenoje;
- gaunamas transportavimo klaidų atsakas iš SIP transporto sluoksnio, kai transakcija pereina į nutraukimo būseną;
- gaunama užbaigimo būseną su 2xx kodo atsakymu iš transakcijos vartotojo, kai transakcija tampa nutraukimo būsenos išsiuntus atsakymą.

Laiko žymeklis H inicijuojamas, kai tampa užbaigimo būsenoje. Laiko žymeklis G inicijuojamas valdyti laiką kiekvienai retransliacijai iš intervalo 300-699 atsako, kuris anksčiau buvo gautas iš transakcijos vartotojo veiklos būsenoje (kada transportavimas yra nepatikimas). Tuomet yra galimybė atsirasti vienam iš šių įvykių:

- gaunama retransliuota INVITE užklausa;
- transakcija retransliuoja 300-699 atsakymus, gautus iš transakcijos vartotojo, esant tyrimo būsenoje;
- laiko žymeklis G inicijuojamas, transakcija retransliuoja intervalo 300-699 atsakus, atstato laiko žymeklį G, lieka toje pačioje būsenoje;
- laiko žymeklis H inicijuojamas, transakcija pereina į nutraukimo būseną;
- ACK žinutė gaunama iš kliento pusės ir transakcija nedaro jokių veiksmų, išskyrus būsenos keitimą į veiklos būseną.

Tuo metu, kai yra veiklos būseną, laiko žymeklis I inicijuojamas ir transakcija laukia ACK trigerio retransliuoti intervalo 300-699 atsakus. Kai laiko žymeklis I sužadinas, nutraukimo būseną įvyksta ir serverio transakcija yra sunaikinama transakcijos vartotojo.



2.2 pav. Sesijos inicijavimo protokolo serverio pusės būsenų mašina (sudaryta autoriaus pagal IETF MMUSIC rekomendacijas)

2.3. Vieningos komunikacijos magistralės sprendimai

SIP yra užklauso ir atsako protokolas ir nepriklausomas nuo transportavimo lygmens. SIP yra praplečiamas metodais, antraštėmis ir žinučių apkrova. Tai leidžia protokolui būti užbaigtu su gausiu standartizuotu praplėtimu, atitinkančiu specifinius poreikius. Čia gali būti integruojami momentiniai pranešimai ir įvykiai. Žinučių keitimasis yra nepriklausomas nuo formato, tai leidžia keistis šioms duomenų rūšims (pvz.: SDP, SOAP).

Vieninga komunikacijos magistralė sujungia įrenginių sesijas. Sujungiant įrenginius SIP protokolu, galimi šie veiksmai: teikti komandas (RPC pagrindu siunčiamus pranešimus), įvykiai ir sesijos duomenų srautai. Šios komunikacijos formos reikalingos koordinuoti įrenginius belaidėse aplinkose. Momentinių pranešimų sąveikos būdas vyksta vienas su vienu. Pavyzdžiui gali būti perduodami sensoriaus-daviklio parametrų duomenys. Įvykio sąveikos būdas yra vienas su daugeliu, tinkamiausias mechanizmas teikti informaciją apie objekto

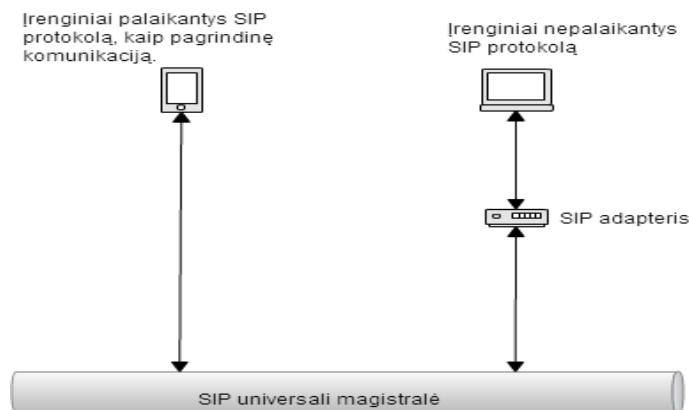
būseną. Sesijos sąveikos būdas yra vienas su vienu duomenų apsikeitimui tam tikru laiku. Dažniausiai naudojame multimediją apsikeitimui tarp dviejų objektų.

SIP suteikia mechanizmą, kuris susijęs su vartotojo mobilumu. Siekdami išspręsti mobilumo problemą, SIP objektai surandami per galinius tinklo adresus naudodamiesi URI (Uniform Resource identifiers) agentais.

SIP platforma yra plačiai naudojama įvairiomis formomis, įskaitant dedikuotas IP telefonijos sistemas ar kitas sistemas ir yra IP telefonijos standartas.

SIP gali būti įterptinis didinant įrenginių ir programinės įrangos sistemų skaičių.

Siūloma naudoti SIP protokolą kaip universalią komunikavimo magistralę. Nagrinėjama, kaip skirtingi nutolę įrenginiai galėtų prisijungti prie vieningos SIP komunikacijos magistralės (2.3 pav.).



2.3 pav. Du galimi įrenginių jungimo prie SIP vieningos magistralės būdai

Kad galėtų komunikuoti skirtingų tipų klasės įrenginiai, reikalingos skirtingos SIP adapterio programinės įrangos komponentės. Kokių jų reikės, priklausys nuo įrenginių tipų. Šiame darbe išskirsime keturis nutolusių įrenginių tipus (žr. 3 lentelė).

2 lentelė. SIP ir adapterio taikymo galimi įrenginių tipai ir jų pavyzdžiai

SIP susiejamų įrenginių tipai		Taikomų įrenginių pavyzdžiai
1	Komunikavimo įrenginiai,	SIP telefonas, IP telefonas, Skype

tipas	naudojantys SIP inter- sąveiką	telefonas, SIP kamera
2 tipas	Komunikavimo įrenginiai, reikalaujantys perprogramavimo	e. kalendorius, išmanusis telefonas (PDA, iphone)
3 tipas	Įrenginiai, neturintys tiesioginių programavimo galimybių	IP kamera, Printeris,
4 tipas	Įrenginiai, nesiejami IP tinklo sąsaja	Sensoriai, ZIGBee, X10

2.4. Keturių nutolusių įrenginių tipų integracijos SIP vieningoje komunikavimo magistralėje sprendimai

Pirmojo tipo įrenginiai, palaikantys SIP protokolą kaip pagrindinę komunikaciją yra suderinti su SIP. Be papildomų įterptinių funkcijų, suteikiamas priėjimas prie SIP funkcionalumo per atitinkamus sąveikos režimus. Norint suteikti prieigą prie SIP subjekto funkcionalumo, apibrėžiami trijų sąveikos režimų prieinamumai: komandos (statusų užklauskos ir subjektų kontrolė), įvykiai (įvykių publikavimas ir įvykių užsakymas) ir sesijos (kvietimai ir duomenų srautas). Sąveikos režimams reikia perduoti ir priimti duomenis, kurie gali turėti skirtingus formatus: komandų parametrų vertės (pvz.: SOAP), įverčių vertės (naudojant XML pagrindų formatą ir sesijos aprašas (SDP aprašas)).

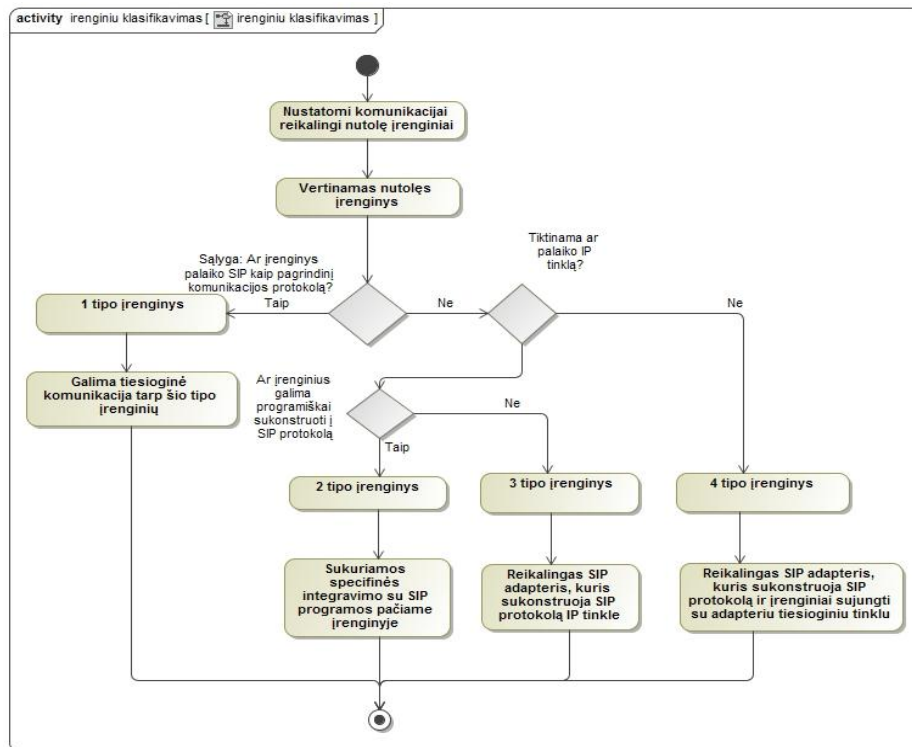
Antrojo tipo įrenginiai – įrenginiai, turintys programinę įrangą, kuri palaiko sesijos inicijavimo protokolą. Šis funkcionalumas leidžia įrenginiui pasinaudoti SIP internetinės telefonijos funkcijomis. Programinė įranga įrenginyje apima visą SIP steką ir integruotas interneto valdymo paslaugas, kurios leidžia sukurti ir priimti SIP žinutes. Antrojo tipo įrenginys yra programuojamas SIP adapteris, kuris yra tame pačiame įrenginyje. (pvz. išmaniajame telefone) ir padaro jį autonominiu.

Adapterio sluoksnis yra atsakingas už įrenginio funkcionalumą trečio ir ketvirtojo tipo įrenginiams nepalaikantiems SIP, programuojamas SIP adapteris užtikrinantis įrenginio ryšį magistralėje (žr. 2.4 pav.).

Trečiojo ir ketvirtojo tipų įrenginiai nėra programuojami, todėl turime sujungti juos su SIP adapteriu, norint šiuos įrenginius integruoti į bendrą magistralę, kurios komunikacija parempta SIP protokolu. Taip pat vartuose gali būti naudojami ir antrojo tipo subjektai, norint pagerinti komunikacijos našumą, sumažinant energijos suvartojimą arba padidinant paslaugos kokybę. Techninės įrangos vartai yra privalomi ketvirtojo tipo įrenginiams, kadangi jie nepalaiko nei IP nei SIP galimybių.

Trečiojo ir ketvirtojo įrenginių tipų funkcionalumas gali būti prieinamas per ad hoc ryšių magistralės, susidedančias iš programinės ryšio magistralės ir susijusios techninės įrangos ryšio magistralės (Chlamtac, 2003).

Pateikti duomenys gali būti tiesiogiai prieinami per registrus, susietus atmintyje. Sukonstruota tarpinė programinė įranga, papildomai susiejanti į įvykio sąveiką skirtingų rūšių programas, leidžianti įrenginiams bendrauti per komandas ir sesijos sąveikos režimus. Komandos sąveikos režime įrenginiams siunčia SIP MESSAGE užklausa, kuri leidžia veikti esančiame įrenginyje. Sesijos sąveikos režime subjektas siunčia SIP INVITE užklausa susitarti dėl sesijos parametrų ir sukurti sesiją su kitu subjektu. Duomenų apsikeitimui tarp dviejų įrenginių naudojamas SOAP formatas, naudojant kSOAP bibliotekas ir transportavimus per SIP užklausa ir atsakų kūnus (body). Pirmiausia reikia atlikti projektavimo darbus, kad kiekvienas įrenginys, kaip komunikavimo įrenginio tipas, būtų suderinamas su SIP.



2.4 pav. Keturių įrenginių tipų nustatymo algoritmas

Pirmojo tipo įrenginių galimybės

Siejant šiuos įrenginius su SIP sąsaja, tenka suprojektuoti įrenginio (žr. 2.5 pav.) vidurinio sluoksnio programinę įrangą (angl. middleware) taip, kad įrenginiai galėtų tiesiogiai bendradarbiauti vienas su kitu SIP nuosavu (angl. native) protokolu. Programos kode vidurinio sluoksnio programinė įranga turi būti aprašyta, kad leistų tiesiogiai bendrauti, jei šie įrenginiai turi pirmojo tipo savybes. Šioje situacijoje leidžiama tiesiogiai jungtis prie egzistuojančios SIP infrastruktūros (pavyzdžiui, tai galėtų būtų Sailfin ar OpenSER serveris) ir subjektų (pvz., SIP telefonai ar SIP vaizdo kameros).

Antrojo tipo įrenginių galimybės

Šiuo metu egzistuoja įvairiausių įrenginių, leidžiančių įdiegti SIP programinės įrangos galimybes, pradedant nuo išmaniųjų telefonų, baigiant elektroninio pašto programine įranga. Mūsų atveju, sąsajos kūrimas būtų java programavimo kalba su SIP galimybes palaikančiu karkasu, sukuriant iškvietimo sluoksnį ir prijungiant įrenginio funkcionalumus prie bendros SIP komunikacijos magistralės. Programavimas atliekamas aukšto lygio kolboje, konstruojant operacijas, kurios leidžia užregistruoti ir iškviešti įrenginius, įgyvendinti ir

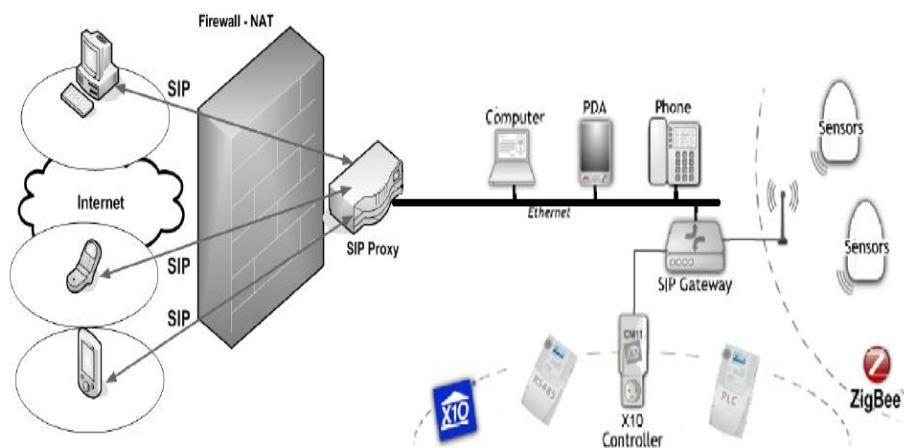
iškviesti subjektų funkcionalumą. Svarbu paminėti, kad taupat antro tipo įrenginių atitinkamas skaičiavimo galimybes ir nenaudojant įrenginių resursų ir padidinant įrenginio našumą reiktų naudoti vartus, kurių našumas daug didesnis.

Trečiojo tipo įrenginių galimybės

Šito tipo įrenginiai yra neprogramuojami, neturintys programinės įrangos, kuri palaikytų SIP, bet palaikantys IP protokolą (pavyzdžiui, tai galėtų būti IP vaizdo kameros). Norint pasiekti šio tipo įrenginius SIP protokolu, reikia suprojektuoti adapterį, kuris konvertuoja įrenginio komunikacijos protokolą į SIP. Reikia sukurti tinklo sietuvą, kad įgyvendintume šito tipo įrenginio sąveikos pritaikomumą tinkle. Norint suprojektuoti tinklo sietuvą siūloma pasirinkti Atmega plokšę su Ethernet lizdu. Atmega plokštė programuojama C kalba. Galima panaudoti GNU OSIP biblioteką, kurią reikia perkelti ir adaptuoti į Atmega plokštę. Taip pat reikia perkelti SIP vartotojo agentą (user agent), kad jis registruotų ir aptarnautų kiekvieną šio tipo įrenginį.

Ketvirtojo tipo įrenginių galimybės

Daugiausia tokių prietaisų sudaro maži įrenginiai ar sensoriai, kurių funkcijos yra apribotos nuskaitant pavienius parametrus. Kadangi jie nepalaiko IP tinklo ir nėra programuojami, jiems tinklo sietuvas yra privalomas.



2.5 pav. Skirtingų tipų nutolusių įrenginių jungimo į bendrą magistralę, taikant SIP pavyzdys

Adapteris turi būti panaudojamas kiekvienam įrenginiui, kuris nepalaiko SIP. 4 tipo įrenginiams adapteris aprašomas žemo lygio komunikacijos operacijomis. Mūsų pateikiamoje sprendimo paramos architektūroje sukonstruotas specifinis adapteris, kuris leidžia nuskaityti sensorių atitinkamus duomenis (tai galėtų būti judesio sensorius, temperatūros sensorius ir kt.) ir yra sujungtas su SIP tinklo sietuvu. Ketvirtojo tipo įrenginiai dažniausiai yra prijungti prie tilto tiesiogiai tam tikromis jungtimis, t.y. laidais.

2.5. SIP protokolo panaudojimas vieningoje magistralėje

Suprojektuota SIP tarpinė programinė įranga, kuri leidžia iškviešti nutolusių įrenginių funkcionalumus, išsiunčiant užklausas ir gaunant atsakus pasinaudojus SIP ryšio magistrale. Ši magistralė taip pat leidžia nuotoliniams įrenginiams bendrauti SIP naudingą apkrovą, t.y. žinutės atitinka visus SIP protokolui keliamus reikalavimus. Ši magistralė palaiko išskirstytų įrenginių suradimo ir informavimo paslaugas. Ši paieškos paslauga leidžia registruoti ir iškviešti prisiregistravusius įrenginius. SIP suteikia galimybę dinamiškai judėti skirtinguose aplinkuose palaikant jų mobilumą. Nutolę įrenginiai siunčia savo esamą adresą pasinaudoję *SIP REGISTER* užklauso žinute. Prisiregistravęs įrenginys gali būti iškviečiamas užklausimais iš registracijos serverio. Dalis įrenginių neturi galimybės prisijungti prie centrinio serverio, kadangi yra sujungti su suprojektuota SIP tinklo tilto privačia magistrale. Privati magistralė - tai sujungimo būdas, kai įrenginys sujungtas IP protokolu.

Duomenų apsikeitimas vyksta pagal scenarijų, kurio veiksmų eigą reglamentuoja SOAP formatas. Čia naudojama kSOAP biblioteka, SIP transportavimo sluoksnis, užklauso ir atsakų aprašas.

Pirmojo tipo įrenginiams mažiausiai reikia papildomų programavimo ir projektavimo primonių, nes šie įrenginiai jau suprojektuoti bendrauti SIP kaip pagrindiniu komunikavimo protokolu, todėl turi reikiamą našumą tokiai komunikacijai užtikrinti.

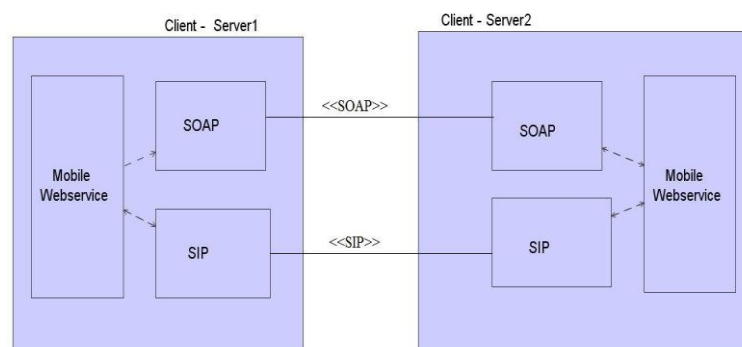
Antrojo tipo įrenginiai dažniausiai turi pakankamą skaičiavimo galią, kad galėtų transformuoti žinutes į SIP standartus atitinkančias žinutes ir palaikytų reikiamus veiksmus pasinaudojus atitinkama programine įranga.

2.6. Žiniatinklio paslaugų integravimo galimybės mobiliuosiuose įrenginiuose

Šiuo metu pagrindinis informacinių sistemų kūrimo uždavinys yra padaryti duomenis prieinamus kuo daugiau žmonių ir kad tuos duomenis atpažintų skirtingi prietaisai, kurie galėtų komunikuoti iš skirtingų duomenų šaltinių. Kiekvienas duomenų šaltinis potencialiai gali būti skirtingose aplinkose (J2EE ar Microsoft .NET). Pasirinkdami skirtingos aplinkos integraciją, leisime skirtingų mobiliųjų dalykinių programų, kurioms sukurti sugaištama daug laiko ir pajamų atsiradimą. Išspręsti integracines problemas buvo sukurtos XML ir žiniatinklio paslaugų technologijos.

XML, SOAP ir WSDL suteikia standartą duomenų mainams tarp platformų. XML aprūpina standartinį duomenų formatą, kurį naudos prietaisai nepriklausomi nuo tinko protokolų ir transportavimo (El-hassan, 2009).

Keičiantis XML duomenimis su nutolusiomis paslaugomis SOAP suteikia standartinės žinutės formatą. Šias technologijas naudojant kartu, siūlomas sprendimas, kaip keistis duomenimis. Žiniatinklio paslaugos yra plačiai adaptuojamos. Taip yra kuriama konkurencinga aukštos kokybės aplinka ir mažai kainuojantys įrankiai, leidžiantys pasiekti duomenis per žiniatinklio paslaugas.



2.6 pav. Duomenų keitimosi tarp dviejų mobiliųjų įrenginių schema taikant SOAP ir SIP

Sukurti žiniatinklio paslaugas nėra sudėtinga naudojant šiuolaikinius įrankius, tad atsiradus naujiems duomenims, juos pasiekti ir apdoroti tampa paprasta. Tai didina duomenų vertę. Duomenys tampa prieinami įvairiems prietaisams, naudojant žiniatinklio paslaugas.

Mažuose mobiliuosiuose prietaisuose XML ir žiniatinklio paslaugos pasirinkimas yra logiškas komunikacijos mechanizmas. Jie naudojami labai retai, kadangi dauguma mobiliųjų prietaisų turi labai mažą dalykinės programos atminties išteklių. Dažniausiai tai būna tik 64 kb ar net mažiau. Paprastai norint XML ir žiniatinklio paslaugas įvykdyti kompiuteriuose, reikia daugiau kaip 20 kartų daugiau atminties, kad galėtume realizuoti reikalingą dalykinę programą. Naudodami žiniatinklio paslaugas mobiliuose dalykinėse programose susiduriame su atminties talpos problema, kurią turime išspręsti.

Mobiliosiose dalykinėse programose žiniatinklio paslaugas panaudoti galime pasirinkę KSOAP ir KXML technologijas, kurioms palaikyti reikia mažiau išteklių. 2.6 pav. parodyta, kokią dalį atminties reikia skirti šioms dalykinėms programoms, kad mobiliosiose dalykinėse programose būtų galima bendrauti žiniatinklio paslaugomis.

Tarnybinės stoties pusės kūrimas yra sudėtingas procesas, kuriam reikalingi įgūdžiai. Tokiu atveju, kliento dalykinės programos, kurdamos tarnybinės stoties paraiškas, sukuria daug etapų, kurie turi būti pakartoti kiekvienai naujai paraiškai (2.7 pav.). Tai servlet struktūros nustatymas: kaip perskaityti prašymus, parašyti atsakymus ir kaip apdirbti galines paslaugas. Visos šitos pasikartojančios užduotys yra J2ME automatizuotos. MIDP prietaisuose, HTTP yra transportinis pasirinkimo protokolas todėl, kad jis yra lankstus, visuotinai pasiekiamas, lengvai sujungtas.

HTTP yra vienintelis ryšio tipas MIDP specifikacijoje ir tokiu būdu jos tinkamumas yra garantuotas. HTTP yra transportinis protokolas, pasirenkamas populiarėms žiniatinklio paslaugų mechanizmams, tokiems kaip XML-RPC ir SOAP. HTTP tinklo susisiekimas yra pagrėstas dalykinės programos ir tarnybinės

stoties modeliu, kur dalykinė programa teikia prašymus ir naudoja atsakymus iš tarnybinės stoties

Labiausiai keičiasi įvairūs paslaugos prašymo mechanizmai, tokie kaip XML-RPC ir SOAP, kurių duomenys yra užkoduoti ir suspaudžiami. XML-RPC yra SOAP pirmtakas, aprūpinantis XML pagrįstomis tolimomis procedūromis. Šiuo metu tenka pasinaudoti trečiosios šalies tolimais paslaugų sprendimais, tokiais kaip kXML-RPC ir kSOAP, kadangi tolimas paslaugos kvietimas dar nėra standartas J2ME platformose. Pateikiami XML gavėjai (*parser*), kurie gali būti panaudojami J2ME platformose:

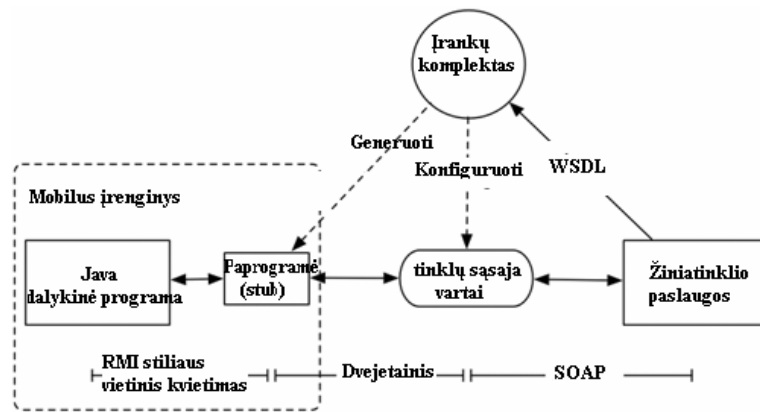
kXML1.2 (apytiksliai reikalauja 16 kb atminties, labai stabilus, geras atminties valdymas)

kXML2 (apytiksliai reikalauja 9 kb atminties, naujesnis, mažas, mažiau išbandytas, bet stabilus, geras atminties valdymas)

NanoXML (apytiksliai reikalauja 10 kb atminties, stabilus, bet intensyvesnė atmintis)

TinyXML (apytiksliai reikalauja 12 kb atminties, stabilus, didesnis atminties pėdsakas)

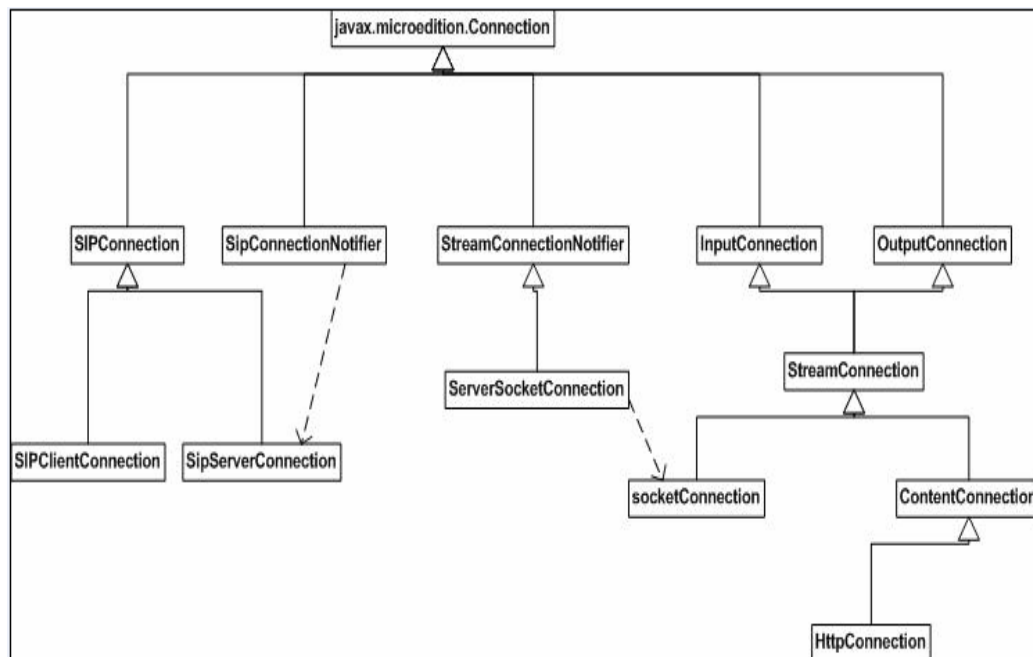
Tolimas paslaugos kvietimo mechanizmas ir XML supratėjas kiekvienoje dalykinėje programoje yra netinkamas sprendimas ištekliams suvaržytuose prietaisuose. JSR172 apibrėžia standartinį tolimo kvietimo metodą XML, J2ME platformose. Belaidžių dalykinių programų projektuotojai labiau mėgsta apsikeisti binariniais duomenimis, o ne XML formato žinutėmis. Žiniatinklio paslaugos mobiliosiose dalykinėse programose (nepaisant vartotojo ir tarnybinės stoties) naudoja XML, bet rekomenduojama naudoti dvejetainių duomenų perkėlimą



2.7 pav. Belaidžių įrenginių sąveikavimo struktūra taikant MIDlet ir WSDL

2.7. SIP programavimo sąsaja mobiliems įrenginiams

Norint pritaikyti SIP technologiją mobiliems įrenginiams, reikia suprogramuoti tam tikras sąsajas. Norint pasiūlyti paslaugas, taikančias SIP protokolo galimybes, reikia pasinaudoti biblioteka (2.8 pav.), kuri papildoma javax.microedition.Connection paketu ir specifiniu javax.microedition.sip paketu, kurie užtikrina sujungimo sesijas tarp SIP klientų



2.8 pav. SIP paslaugų praplėsto paketo klasių diagrama

SIP programavimas galimas su J2ME. Susipažinsime su sąvokomis, kaip atidaryti ir užbaigti kliento ir serverio ryšius, duomenų siuntimo srautus. Taip pat

susipažinsime su klasėmis, kurios yra būtinos, kad galėtume panaudoti SIP technologija. Pagrindinės programuojamos sąsajos klasės aprašytos 1 lentelėje.

3 lentelė. SIP sąsajai užtikrinti reikalingos klasės

Klasė	Apibūdinimas
Connector	Sukuria visų objektų ryšių sujungimus. SIP ryšiams naudojamas adresas, kuris prasideda „sip“.
SipClientConnection	Ši klasė naudojama, norint išsiųsti SIP žinutes, kurios nėra pasikartojančios.
SipClientConnectionListener	Ši sąsaja turi būti įgyvendinta klasėse, kurios apdoroja SIP atsakymus.
SipServerConnectionListener	Ši sąsaja turi būti įgyvendinta klasėse, kurios planuoja gauti SIP užklausas.
SipServerConnection	Ši klasė inicijuoja sesiją įeinančioms žinutėms.
SipRefreshHelper	Klasė, kad valdytų grįžtančias išeinančias SIP žinutes, kaip registras ir užsakymas.
SipRefreshListener	Įgyvendina sąsają, kad apdorotų atsakymus grįžtančioms žinutėms.

2.8. Antrojo skyriaus išvados

Išnagrinėtas SIP protokolo veikimas, kuris leido integruoti skirtingus nutolusius įrenginius ir juos jungti į vieningą komunikavimo magistralę.

SIP veikimas aprašomas daugiasluoksne architektūra, kurios pagrindinis veikimo principas grindžiamas užklausų ir atsakų ryšiais tarp įvairių tinklo komponentų.

Aprašytas Sesijos inicijavimo protokolas, kuris suskirstytas į keturis sluoksnius, kurių kiekvienas vykdo atitinkamus funkcijų rinkinius. Klasifikuojant nutolusių įrenginių veikimo ir integravimo ypatybes bei galimybes ir sujungiant

tinklą, reikalinga papildoma adaptavimo įranga. Atsižvelgiant į šios klasifikacijos ypatumus ir taikomus skirtingus įrenginių jungimo metodus pateikta metodika, kaip parinkti tinkamas adapterio funkcijas ir sukurti programinį palaikymą, užtikrinantį vieningos SIP komunikacijos magistralės darbą.

3. PETRI TINKLŲ TAIKYMAS SIP KOMUNIKACIJOS PROCESAMS APRAŠYTI IR MODELIUOTI

Šiame skyriuje aprašoma spalvotųjų Petri tinklų pritaikymas SIP funkcinė savybių aprašymui, modeliavimui ir analizei. Pristatomos imitacinio modeliavimo priemonės – spalvotieji Petri tinklai (CPN) nutolusių mobilių objektų stebėjimo scenarijams imituoti ir SIP veikimo procesams modeliuoti. Spalvotieji Petri tinklai leido stebėti sesijos inicijavimo protokolo skirtingų situacijų vykdomas būsenas konkrečiais laiko momentais ir judančių duomenų srautų judėjimą, operacijų vykdymo sąlygas. Sukurti modeliai leido stebėti perduodamos informacijos struktūrą, kuri reikalinga įvertinti keičiantis SIP būsenoms, bei komunikavimo skirtingomis sąlygomis.

3.1. Petri tinklų formalus aprašymas

C. A. Petri savo disertacijoje 1962 metais pasiūlė modeliavimo tinklus kaip grafinį formalizmą, leidžiantį aprašyti, modeliuoti ir valdyti sudėtingus, lygiagrečius, asinchroninius ir sinchroninius procesus. Šie tinklai labai išpopuliarėjo sėkmingai taikant juos skaičiavimo sistemų aprašymui ir įgijo Petri tinklų pavadinimą. Petri tinklai sparčiai vystomi ir dabar, kuriami standartai, jų formalizavimo ir modeliavimo priemonės taikomos daugelyje probleminių sričių, tokių kaip traukinių eismo valdymas, komunikacinių tinklinių sistemų darbo valdymas, automatizuotų valdymo sistemų, bendradarbiavimo, kompiuterinių agentų planavimas ir valdymas.

Petri tinklus formaliai galima apibrėžti keturių aibių pagalba:

$$N = (B, A, \Phi, H),$$

kur B yra baigtinė aibė simbolių, vadinamų pozicijomis $B \neq \emptyset$; A - baigtinė aibė simbolių, vadinamų perėjimais, $A \neq \emptyset$; $B \cap A = \emptyset$, $\Phi: B \times A \rightarrow \{0, 1\}$ yra tiesioginė incidentiškumo funkcija;

$H: A \times B \rightarrow \{0, 1\}$ - atvirkštinė incidentiškumo funkcija.

Nurodytos keturios aibės nusako Petri tinklo struktūrą. Paprastai Petri tinklas atvaizduojamas dviskilčiu grafu, kurio viršūnių aibė yra $B \cup A$. Pozicijų viršūnės

atvaizduojamos rutuliukais, perėjimų pozicijos – vertikaliais brūkšniais. Iš pozicijos b_i eina lankas į viršūnę - perėjimą, tada ir tik tada jei $\Phi(b_i, a_j)=1$; iš perėjimo a_j į poziciją b_i eina lankas, tada ir tik tada, jei $H(b_i, a_j)=1$. Kiekvienam perėjimui $a_j \in A$ galima apibrėžti aibę įėjimo pozicijų $\Phi(a_j)$ ir aibę išėjimo pozicijų $H(a_j)$:

$$\Phi(a_j) = \{b_i \in B \mid \Phi(b_i, a_j) = 1\}$$

$$H(a_j) = \{b_i \in B \mid H(a_j, b_i) = 1\}$$

$$\text{kur } i = \{1, \dots, n\}; \quad j = \{1, \dots, m\}; \quad n = |B|; \quad m = |A|.$$

Analogišku būdu įvedama aibė įėjimo perėjimų pozicijai b_i - $H(b_i)$ ir aibė išėjimo perėjimų $\Phi(b_i)$ pozicijai b_i :

$$H(b_i) = \{a_j \in A \mid \Phi(a_j, b_i) = 1\}$$

$$\Phi(b_i) = \{a_j \in B \mid \Phi(b_i, a_j) = 1\}$$

Duotas Petri tinklų apibrėžimas gali būti pritaikomas tik modeliuojamos sistemos statikos (t.y. įvykių tarpusavio ryšių ir sąlygų) atvaizdavimui. Tam, kad galima būtų atvaizduoti funkcionavimo dinamiką, įvedama tinklo pažymėjimo, žymės (markerio) funkcija:

$$M: B \rightarrow \{0, 1, 2, \dots, n\}.$$

Remiantis funkcija M tinklo pozicijos žymimos sveikaisiais teigiamais skaičiais, kurios grafiškai atvaizduojamos taškais talpinamais pozicijose ir vadinamais žymėmis.

Tinklas funkcionuoja pereinant nuo vieno pažymėjimo prie kito pagal perėjimo apibrėžtą pozicijų sujungimo struktūrą. Pradinis tinklo žymėjimas aprašomas: $M_0 = B \rightarrow \{0, 1, 2, \dots, n\}$. Žymių pasikeitimas įvyksta vieno iš perėjimo $a_j \in A$ suveikimo rezultate. Būtina perėjimo a_j suveikimo sąlyga yra $\forall b_j \in \Phi(a_j) \{M(b_i) \geq 1\}$, kur $M(b_i)$ yra pozicijos b_i pažymėjimas.

Perėjimas a_j , kuriam išpildoma duota sąlyga, apibrėžiamas kaip aktyvus. Aktyvaus perėjimo a_j suveikimas pakeičia tinklo pažymėjimą $M(b) = (M(b_1), M(b_2), \dots, M(b_n), M(b_N))$ į pažymėjimą $M'(b)$ pagal taisyklę $M'(b) = M(b) - \Phi(a_j) + H(a_j)$. Tai yra, perėjimas a_j išima po vieną žymę iš savo kiekvienos įėjimo pozicijos ir patalpina po vieną žymę į kiekvieną savo išėjimo poziciją. Žymėjimo

pasikeitimui iš M į M' tinkle pavaizduoti naudojama išraiška $M' \xrightarrow{a}_j M$. Petri tinklas, kuriame naudojama žymėjimo funkcija, vadinamas pažymėtu Petri tinklu ir yra aprašomas sąryšiu $N=(B, A, \Phi, H, M_0)$, kur M_0 yra pradinis tinklo pažymėjimas. Labai svarbi Petri tinklų savybė yra modelio sudarymo hierarchinių sudėtingumo lygių atvaizdavimo galimybė. Kiekvienas tinklas gali būti suprantamas kaip makro perėjimas arba makro pozicijų. Be to, perėjimas ar pozicija gali būti detalizuoti kaip atskiri potinkliai.

Petri tinklai pagal formalų apibrėžimą, pateiktą [Petri,1960], tinka nedeterminuoto laiko intervalo įvykiams aprašyti. Šiuo atveju modelis išreiškia tik įvykių pasireiškimo momentus. Petri tinklo funkcionavimas vyksta nedeterminuotai tada, kai susidaro sąlygos perėjimo aktyvizavimui (t.y. atsiranda pažymėjimas, patenkinantis vieno ar kelių perėjimų paleidimo sąlygas). Vieno perėjimo suveikimas tokia modelyje laikomas momentiniu, nes laikomasi prielaidos, kad tikimybė vienu metu įvykti dviems ar daugiau įvykių lygi nuliui. Ši Petri tinklų savybė, neišreiškianti laiko parametrų panaudojimo, modeliuojant determinuotus procesus, yra esminių jų taikymų, modeliuojant skaičiavimo sistemas, apribojimų priežastimi.

Naudojant Petri tinklus pagalba sudarytas modelis leidžia tirti labai svarbias sistemos veikimo savybes:

S1. *Pasiekiamumą*. Duotame tinkle N_M , kurio pradinis žymėjimas M_0 , reikia nustatyti: ar pasiekiamas pažymėjimas M' iš pradinio duoto pažymėjimo M_0 . Ši modeliavimo problema interpretuojama kaip galimybė pasiekti atitinkamą sistemos veikimo būseną. Analizės metu panaudojama visų pasiekiamų tinklo N_M pažymėjimų aibės sąvoka. Ši aibė žymima $R(N_M)$.

S2. *Gyvybingumą*. Petri tinklo perėjimo gyvybingumo savybė nusako principinę galimybę, kad perėjimas suveiks, esant tam tikram pradiniam tinklo pažymėjimui M_0 . Petri tinklas patenkina gyvybingumo savybę, jeigu jo visi perėjimai yra gyvybingi. Duoto modelio gyvybingumo analizė leidžia nustatyti negyvybingus perėjimus, t.y. negalimas modeliuojamos sistemos būsenas.

S3. *Tinklo pažymėjimo apribojimus*. Pozicija b_i tinkle N_M vadinama apribota skaičiumi k , jei egzistuoja toks sveikas skaičius k , kuriam $M(b_i) \leq k$, bet kuriam

tinklo pažymėjimui $M \in R(N_M)$. Jeigu tinkle N_M , $\forall b_i \in B \{M(b_i) \leq k\}$, $\forall M \in R(N_M)$, tai N_M vadinamas k apribotu.

S4. *Tinklo apsaugojimą.* Tinklas N_M vadinamas apsaugotu, jei tenkinama sąlyga:

$$\forall M(b_i) \in R(N_M) \{ \forall b_i / M(b_i) \leq 1 \}, i = \{1, \dots, n\}; n = |B|.$$

Apsaugotu vadinamas toks Petri tinklas, kurio bet kurioje pozicijoje, esant bet kokioms sąlygoms, negali būti daugiau nei viena žymė (t.y. tinklo apribojimas k lygus 1).

Realiųjų sistemų modelių ribojimo ir apsaugojimo savybių analizė leidžia nustatyti sistemų funkcionavimo galimybes esant tam tikriems stacionariems režimams. Šių savybių tyrimų metu, pvz., gali būti nustatyti reikalavimai sistemos buferinėms atmintims ir pan.

S5. *Tinklo pažymėjimų išlaikymo savybė.* Kita tinklo N_M pažymėjimų apribojimo sąlyga gali būti išreiškiama:

$$\sum_{b_i \in B} M(b_i) = const, \text{ kiekvienam } M(b_i) \in R(N_M),$$

$$b_i \in B$$

arba papildomai priskiriami svoriai w_{bi} pozicijoms, ir tada ši sąlyga išreiškiama štai taip:

$$\sum_{b_i \in B} M(b_i) w_{bi} = const, \text{ kiekvienam } M(b_i) \in R(N_M),$$

$$b_i \in B$$

Tinklo pažymėjimų išlaikymo savybės analizė yra svarbi, kai tiriami judantys objektai (žymės) atvaizduoja nekintančius sistemos resursus.

Įprastų Petri tinklų panaudojimo patirtis parodė, kad didelio matavimo tinklo gyvybingumo, pasiekiamumo ir pan. savybių analizė reikalauja daug imlaus darbo. Ši priežastis paskatino įvesti tam tikrus ribojimus tinklo struktūroms, kurie supaprastintų jų analizę.

Įvestas ribojimas:

$$\forall a_j \in A \{ \Phi(a_j) = H(a_j) = 1 \}; j = \{1, \dots, m\}; m = |A|$$

leidžia priskirti Petri tinklus grafams automatams.

Pozicijų įėjimo ir išėjimo perėjimų aibės ribojimas

$\forall b_j \in B \{ \Phi(b_j) = H(b_j) = 1 \}; j = \{1, \dots, n\}; n = |B|$ yra pažymėtų grafų poklasio požymis.

Išvystant Petri tinklų modeliavimo galimybes, buvo pasiūlyta daug jų praplétimo variantų:

- *Apibendrinti Petri tinklai:*

$N_G = (B, A, \Phi, H, M_0)$, kuriems galioja

$\Phi: B \times A \rightarrow w_\Phi = \{0, 1, 2, \dots\};$

$H: A \times B \rightarrow w_H = \{0, 1, 2, \dots\};$

Apibendrinančio tipo Petri tinklo grafas turi daugkartinio grafo formą, kuriame galimi keli lankai tarp pozicijos ir perėjimo. Šie lankai gali būti pažymėti tam tikrais svoriais. Tačiau toks Petri tinklų praplétimas nepadidina modeliuojamųjų savybių ir iš principo apibendrinti tinklai visiškai ekvivalentūs įprastiems Petri tinklams.

Nuspalvintieji Petri tinklai taip pat savo galimybėmis ekvivalentūs įprastiems tinklams, tik čia įvedama dar viena žymių nuspalvinimo aibė, kuri leidžia sumažinti sudėtingos sistemos funkcionavimą atvaizduojančio grafo struktūros sąskaita.

Prioritetiniai ir nulio patikrinimo tinklai leidžia įvertinti modeliuojamos sistemos įvykių prioritetiškumą. Tuo tikslu, nulio patikrinimo tinkluose papildomai įvedama aibė Φ tiesioginių incidentiškų uždraudimo lankų, kuriems $\Phi \cap \Phi_I = \emptyset$.

Grafe lankai, atvaizduojantys aibę Φ_I , vadinami uždraudimo lankais. Tokio tipo tinkle perėjimas suveiks išpildžius dvi sąlygas:

- visose perėjimo įėjimo pozicijose $\Phi(a_j)$ turi būti po žymę;
- visose uždraudimo pozicijose $\Phi_I(a_j)$ neturi būti žymių.

Prioritetiniuose Petri tinkluose įvedama speciali prioritetiškumo funkcija, aprašanti perėjimų paleidimo prioritetus.

Sudarant modelį, dažnai tenka įvertinti modeliuojamų įvykių laiko parametrus. Tuo tikslu buvo pasiūlyti Petri tinklų praplétimai: laiko tinklai, Merlino tinklai, E-tinklai, spalvotieji Petri tinklai (CPN).

3.2. Spalvotieji Petri tinklai (CPN)

Spalvotieji Petri tinklai (CPN) leidžia nagrinėti sistemą gana detalai, nes kiekvienas sistemos žingsnis aprašomas atskirais komponentais. Kiekviename paslaugų sistemos projektavimo žingsnyje analizuojama reali situacija. Tinkamam realizaciniam algoritmui sukurti taikomas atitinkamas Petri tinklų architektūros modelis, pagal kurį vykdomas paslaugos atlikimo scenarijus. Kiekviena sistemos posistemė suskaidoma į komponentus – operacijas ir būsenas. CPN modeliuoja sistemos būsenas. Ryšių kanalais perduodama informacija, reikalinga keičiant vieną būseną kita, jais atliekami valdančios informacijos perdavimai ir / arba materialiujų srautų judėjimai. Sistemos būseną nustatoma "būsenos žymėmis", ji fiksuojama tam tikroje pozicijoje. Būsenų pokyčiai vaizduojami žymių pokyčiais. Žymių srautai atitinka objektų parametrų pokyčius (išteklių, signalų, duomenų) ir modeliuojami sistemoje per žymių parametrų aibes bei operatorius.

Spalvotieji Petri tinklai (CPN) apibūdinami tokiu aibių rinkiniu (Billington et al. 2004):

$$CPN = (\Sigma, P, T, A, N, C, G, E, I);$$

čia Σ – baigtinė, ne tuščia spalvų aibė; P – baigtinė pozicijų aibė; T – baigtinė perėjimų aibė; A – baigtinė, ne tuščia jungčių aibė; N – mazgų (transformacijų, funkcijų) aibė, siejanti kiekvieną jungtį su pora (pozicija, perėjimas) arba (perėjimas, pozicija); C – pozicijų spalvų funkcija, siejanti kiekvieną poziciją iš aibės P su spalva iš aibės Σ ; G – perėjimų kontrolės reiškinių (funkcijų) aibė; E – jungčių reiškinių (funkcijų) aibė; I – inicijavimo reiškinių (funkcijų) aibė. Aibių elementai žymimi mažosiomis raidėmis. M - tinklo žymėjimą realiuoju laiku nusakanti multi-aibė, apimanti pozicijose esančius žymenis, o Y - einamuosius tinklo žingsnius nusakanti aibė. Petri tinklo veikimą galima nusakyti pagal perėjimų suveikimo principus (Ding, Liu, 2008), kur pradinis tinklo žymėjimas aprašo galimybę iš pradinės būsenos pereiti į kitas būsenas: $\forall p \in P : M_0(p) = I(p)$; $n+1$ -asis tinklo žymėjimas: $\forall p \in P : M_{n+1}(p) = I(p)$.

Dažnai norima atvaizduoti objektų parametrus (pvz., pavadinimą, kiekį). Tokių parametrų sudėtingesniems sąveikos būdams atvaizduoti įvedamos

spalvotos žymės. CPN modelyje parametro tipą nurodo spalva. Perėjimai aprašomi taisyklėmis, kurių pagalba keičiamos generuojamų žymių spalvos. Panaudotų žymių spalvomis aprašomi ryšiai tarp įėjimo žymių ir išėjimo žymių. Taip pat galima aprašyti „prieš sąlygas“, kurios nustato naudojamų žymių spalvas ir pagal tai atpažįsta parametrų tipus bei galimas operacijas atliekamas su šiais parametrais. Spalvoti Petri tinklai suteikia galimybę žymėti skirtingomis spalvomis tiek procesus, tiek ir vykstančių parametrų pokyčius ir tuo būdu atspindėti sutartines duomenų reikšmių dinamines savybes. Kiekviena pozicija gali būti susieta su tam tikrų parametrų tipų rinkiniu, kuris apsprendžia duomenų tipų rinkinio struktūrą, kurią pozicija gali saugoti ir yra toje pozicijoje išreiškiamų sąlygų įvykdymo išraiška.

CPN modelis konstruojamas taikant pozicijų (angl. places), vaizduojamų apskritimais, ir perėjimų (angl. transitions), vaizduojamų stačiakampiais, bei nukreipimo lankų grafinę notaciją. Sąveika modeliuojama kryptinėmis linijomis ir imituoja darbų, valdymo arba duomenų perdavimo srautus. Modelis yra aprašomas CPN ML programavimo kalba. Pozicijose esamos žymės atvaizduoja sistemos būklę. Kiekvieną poziciją galima pažymėti viena ar daugiau žymių ir kiekviena žymė neša savyje duomenų reikšmę, įgyjamą modeliuojamu momentu.. Tai duomenų žymės spalva. Realią duomenų sistemos būseną nusako žymių skaičius ir žymių spalvos esančios tam tikrose pozicijose, kurios tuo pačiu metu ir atvaizduoja sistemos būseną. Tai vadinama CPN modelio žymėjimu. Šalia kiekvienos pozicijos yra aprašas, pagal kurį nustatomas spalvų rinkinio požymis (duomenų vertė). Spalvų rinkinys yra apibrėžiamas naudojant CPN ML raktažodį - „colset“, o spalvos vertė – NO – gali būti apibrėžiama visomis sveikųjų skaičių INT reikšmėmis.

CPN leidžiama atlikti interaktyviąją imitaciją, kurios metu rezultatai pateikiami tiesiogiai diagramoje. Imitaciniame modelyje galime lengvai atvaizduoti didelę ir sudėtingą sistemą ir stebėti jos veikimą. Imitaciniame CPN modelyje galima stebėti ir reikalui esant keisti judančių žymių „pernešamą“ informaciją. Taip pat leidžiama atlikti sistemos būsenos analizę, t.y. nustatyti konfliktuojančias būsenas, konkurencinius veiksmus, sinchronizacijos problemas,

pasikartojančių būsenų ir bendrų resursų panaudojimo problemas. Norint fiksuoti gaunamų parametrų sekas belaidžio tinklo protokole, įvedami papildomi parametrai, numatomi SPN aprašyme: colset - spalvų rinkinys, NO = int - spalvą atitinkančio parametro tipo numeris. NO naudojamas modeliuoti veiksmų sekos eiliškumą protokole.

Spalvų rinkinio ir duomenų Dekarto sandauga NOxDATA išreiškia gaunamų duomenų ir jų prieskyrų skirtingiems duomenų tipams sąveiką. Čia pirmas narys yra pozicijos eilės numeris, antrasis narys – fiksuojami duomenys išreiškiami teksto eilute.

Pagal (Homepage of the CPN Tools, 2011) spalvų rinkiniai aprašomi pagrindiniais parametrais taip:

colset DATA = string;

colset NOxDATA = product NO * DATA.

Spalvos rinkinys DATA yra naudojamas tam, kad modeliuotų naudingą duomenų paketų apkrovą ir yra apibrėžiamas teksto eilutės (String) rinkiniu. Spalvų rinkinys NOxDATA naudojamas modeliuoti duomenų paketus, kur taikomi duomenys yra eilės numeris ir patys duomenys.

3.3. SIP veiklos procesų modeliavimo ir imitavimo galimybės taikant Spalvotuosius Petri tinklus (CPN)

Ne visais atvejais galime stebėti sudėtingų sistemų veikimą realybėje, nes tai yra per sudėtinga. Todėl naudojamos imitacinio modeliavimo priemonės. Taikant imitacinius modelius nereikia atlikti eksperimentų realybėje, nes atliekami virtualūs eksperimentai. Imitacinių modelių naudojimas sumažina sprendžiamos problemos kainą, sutaupo laiko, padarytos klaidos kainuoja ženkliai mažiau.

Kurti imitacinius modelius pasirinkti spalvotieji Petri tinklai ir pasirinktas Danijos Aarhus universiteto sukurtas spalvotųjų Petri tinklų modeliavimo įrankis CPN Tools ir Design/CPN. Šis įrankis leidžia konstruoti spalvotuosius Petri tinklus, vykdyti imitaciją, tikrinant perėjimų tipus ir modelio sintaksę. Įrankiai spalvotuose Petri tinkluose leidžia būsenų erdvių analizavimą (t.y. tikrinti būsenų

pasiekiamumą) . (DING, 2008) Spalvotieji Petri tinklai (Colored Petri Net, toliau – CPN) taikomi didelėms ir sudėtingoms kompiuterių sistemoms modeliuoti ir analizuoti. CPN grindžiami intuityviu grafiniu atvaizdavimu. Jie yra įvykdomi, gali būti konstruojami hierarchiniu modeliavimo principu. Juose numatyta galimybė modeliuoti įvairių sistemos procesų sugaišto laiko sąnaudas ir panašius procesų vykdymo parametrus. CPN yra įrankis, kuriuo patvirtiname diskrečius sistemos įvykius, t. y. tiriamus ir modeliuojamus procesus. CPN gali būti naudojami reikiamai informacijai gauti iš išorinių sistemai duomenų struktūrų, analizuoti ir taikyti kaip kontekstinę informaciją vidinių dinaminių modeliuojamų sistemų valdymui (informavimui) (Liu, 2009). Grafinės CPN savybės patikslina taikymų galimybes ir teikia tinkamas priemones daugeliui modeliuojamos sistemos lygių vizualizacijai (Bai, 2011). Be to, sinchroniniai ir asinchroniniai įvykiai gali būti organizuojami ir valdomi pagal tam tikrus prioritetus, nustatant procesų vykdymo eiliškumą, įvairias struktūrines procesų konfigūravimo savybes, procesų įvykdymo sąlygų taikomuosius efektus (Zhu, 2010). Pagrindinis CPN ir Petri tinklų skirtumas yra tas, kad CPN elementai yra atskiriami tam tikrų spalvų priemonėmis, kurios papildo tradicinių Petri tinklų procesų valdymo savybes .

Spalvotųjų Petri tinklų būsenų erdvė naudotina tikrinant (verification) nagrinėjamos sistemos savybes. Spalvotuosius Petri tinklus tikslinga naudoti kaip UML sekų ir klasių diagramų papildymą, jei nežiūrėsime į tai, kad jie netinkami sudarant sistemų architektūros ir struktūros statinį aprašą (Hong-xia, 2009). Pagrindinė spalvotųjų Petri tinklų ypatybė yra ta, kad jais galime modeliuoti skirtingas sistemų klases, t.y. galima modeliuoti plataus diapazono konkurentines ir išskirstytas sistemas (Gehlot, 2006).

Spalvotųjų Petri tinklų modeliavimo kalba leidžia aprašyti dideles ir sudėtingas sistemas (Liu, 2011). Pagrindinės priežastys, kodėl reikėtų pasirinkti spalvotuosius Petri tinklus yra šios:

1. Spalvotaisiais Petri tinklais sukūrus modelį esti galimybė praktiškai tyrinėti dar nerealizuotą sistemą, t.y. modeliuojamas sistemos aprašas gali būti panaudotas reikalavimų ar būsimos sistemos pristatymo tikslais.

2. Modelių veikimas gali būti analizuojamas daugiakartinėmis imitacijomis.

3. Gauname daugiau informacijos, geriau suprantame modeliuojamą sistemą, darant sistemos aprašą ir analizę.

4. Spalvotieji Petri tinklai – atvaizduojami grafiškai, kurie yra patrauklūs neturintiems pakankamai žinių apie šio tipo tinklus. Kita priežastis – šie tinklai yra panašūs į daugumą neformaliųjų brėžinių, kuriuos projektuoja inžinieriai ar projektuotojai.

5. Spalvotuose Petri tinkluose aiškiai apibrėžta semantika, nusakanti tikslų sistemos veikimą.

Spalvotieji Petri tinklai turi aiškų būsenų ir veiksmų aprašą, skirtingai nuo kitų kalbų, galinčių aprašyti tik būsenas ar tik veiksmus. Spalvotieji Petri tinklai apima valdymą, sinchronizavimą, manipuliavimą duomenimis. Priešingai nuo kitų kalbų, kuriose modeliuojama veiksmų aplinka, o detalės nurodomos atskirai, čia bendroje erdvėje modeliuojama ir aplinka, ir veiksniai, ir rezultatai. Spalvotuose Petri tinkluose yra įvestas laikas, kur be duomenų turi galimybes stebėti laiko trukmės parametrus. Taip pat šiais tinklais galima atlikti interaktyviąją imitaciją, kurios metu rezultatai tiesiogiai pateikiami diagramoje. Galima matyti kaip keičiasi sistemos atitinkamos būsenos ir kaip "pernešami" duomenys. Modeliavimo aplinka taip pat turi kompiuterinių įrankių, leidžiančių analizuoti, imituoti ir konstruoti sudėtingus sistemų veikimo . Norint tinkamai suprogramuoti scenarijus, reikia atlikti nemažai darbų ir klaidų tikimybė yra labai didelė. Be to, klaidos kainuoja labai brangiai. Pasitaiko, kad klaidų neįmanoma atrasti, be to, programuojant scenarijus sudėtingose sąveikiuose sistemose, reikia surasti tinkamus tokių sistemų modelio kūrimo būdus, kad būtų galima tinkamiau valdyti heterogeninių įrenginių komunikavimo ir stebėsenos procesus.

Siūloma tokioms sistemoms aprašyti, modeliuoti bei kontroliuoti taikyti spalvotuosius Petri tinklus. Tyrinėjami spalvotieji Petri tinklai leido atvaizduoti skirtingus projektuojamos sudėtingos sistemos detalumo lygmenis, naudojant

lengvai suprantamas grafiškai atvaizduojamų objektų savybes, atspindinčias ir procesų dinamikos išraiškos priemones.

Darbe keliamas eksperimentinio tyrimo uždavinys sumodeliuoti galimas SIP būsenų stebėjimo situacijas ir šiuos modelius integruoti į sistemos žinių bazę, kad šie modeliai būtų panaudoti sprendžiant dėl tolesnių sprendimų paramos veiksmų valdymo klausimus. Imitaciniai modeliai turėtų padėti lengviau atvaizduoti realiai veikiančių objektų elgseną atspindinčius procesus. Modeliui kurti pasirinkta spalvotųjų Petri tinklų projektavimo metodologija (Lianzhang, 2009). Šie tinklai yra Petri tinklų formalizmo praplėtimas, leidžiantis atvaizduoti realybėje vykstančių procesų bei jų parametrų kaitą, ją stebėti ir valdyti (Dzemydienė, 2010). Sukurti modeliai turėtų būti taikomi belaidžių įrenginių infrastruktūroje integruojant juos į bendrą judančio objekto stebėsenos ir būsenos nustatymo bei atpažinimo sistemą, kaip vieną iš žinių bazės komponentų, praplečiant sistemos, pateiktos (Dzemydienė, 2010), architektūrą ir įtraukiant į ją naujas funkcines savybes.

Straipsnyje (Ding, 2008) ištyrinėtos ir aprašytos sesijos inicijavimo protokolo savybės darant prielaidą, kad transporto terpė, per kurią protokolas veikia, yra saugi ir patikima.

Analizuojant SIP, apžvelgiamos nepatikimoje terpėje jo savybės, dėl kurių gali persitvarkyti ar būti prarandami pranešimai (Usui, 2009). Šiuo metu vykdomi, plėtojami ir palaikomi šio protokolo darbai IETF koncerno. Sesijos inicijavimo protokolo modeliavimas ir analizavimas naudojant formalius metodus aprašyti RFC 3261 specifikacijoje, kuri nusako ar turinys yra teisingas.

Darbe naudosime spalvotuosius Petri tinklus (CPN) kaip modeliavimo ir analizavimo techniką, tikrinant komunikacijos protokolus. Spalvotieji Petri tinklai taip pat turi programinės įrangos paketą (CPN Tools).

Gehlot straipsnyje (Gehlot, 2010) sumodeliuotas paieškos protokolas SIP pagrindu daugiakanaliais servisais orientuotai architektūrai. Šis protokolas nenaudoja Invite transakcijos kaip vienos iš pagrindinių komponentų mobilioje aplinkoje teikiančių interneto paslaugas. Straipsnyje autoriai modeliuoja SIP apsaugos mechanizmus. Analizuota CPN būsenų erdvės atakos scenarijuje.

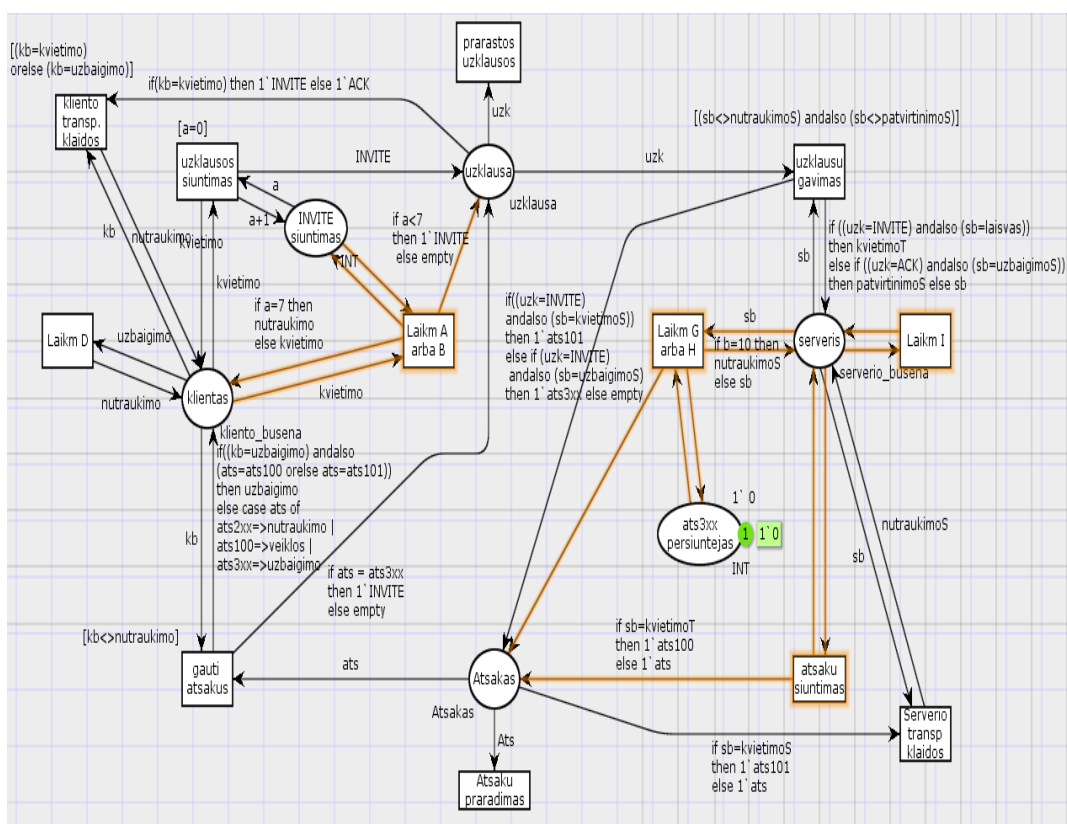
Sekantis susijęs darbas, aprašytas Peng et al. 2007) ir (Sun, 2007). Vykdomos SIP užduotys per operacijų eilę. Pagrindinės dvi SIP operacijos yra *INVITE* operacija susiejanti sesijas ir kitos ne Invite operacijos išsaugančios ir uždarančios sesijas. Pagal (Rosenberg, 2002) SIP Invite operacija gali veikti patikimai (pvz.: *TCP*) arba nepatikimai (pvz.: *UDP*) transportavimo terpėje (Tanna, 2006). Pagal (Hentehzadeh, 2011; Hlavacs, 2008) siūlomą patikimą transportavimo terpę.

3.4. INVITE transakcijos modeliavimas CPN priemonėmis

Duomenys, kurie nerodomi būsenų mašinoje, bet yra susiję su transakcijos operacijomis, turi būti aprašomi, tai yra sąryšiai tarp skirtingų laiko žymeklių. Pirmiausia reikia aprašyti šiuos aspektus ir tik tada įvedinėti INVITE transakciją į CPN modelį nepatikimoje terpėje.

INVITE transakcija privaloma pirminė skambinimo būsenai kliento pusėje, prieš gaunant INVITE užklausą transakcijos vartotojo. Tačiau transakcijos vartotojas gauna INVITE iš transakcijos vartotojo į SIP transporto sluoksnį dar iki atsirandant skambinimo būsenai, kas yra neįmanoma. Tuo metu, kai tik kliento transakcija yra sukurta ir yra skambinimo būsenoje, INVITE transakcijos gavimas iš transakcijos kliento yra įmanomas. Transakcijos vartotojas sugeneruoja atsakymą per 200ms. Serverio transakcija turi sugeneruoti ir išsiųsti *TRYING* (reikšmė 100) atsaką po užklausos sukūrimo. Serverio transakcija tampa *TRYING* būsenoje, kai gauna *INVITE* užklausą ir išsiunčiamas *TRYING* atsakymas. Transakcija tampa šioje būsenoje iškart, kai tik *INVITE* užklausa sukurama ir gaunama iš transakcijos vartotojo. Vienintelis įvykis galintis būti šioje būsenoje yra *Trying* atsakas, siunčiamas serverio transakcijos. Pagal 3.1 pav. matoma, kad kai kliento transakcija yra užbaigimo būsenoje, galimi atsakai su reikšmėmis intervale 300-699. Serverio transakcija gali siųsti intervalo 300-699 statuso atsakymą tik kai išsiunčiamas intervalo 100-199 atsako kodas. Tačiau nepatikimoje terpėje galima pertvarkyti žinutes taip, kad 1xx atsakymai galimi kliento pusėje ir po intervalo 300-699 atsako. Taigi įmanoma kliento transakcijai gauti intervalo 100-199 atsakus, kai yra užbaigimo būsenoje. Kai serverio

transakcija yra užbaigimo būsenoje, laiko žymeklis G inicijuojamas, remiantis RFC 3261 specifikacija, laiko žymeklis G gali būti atnaujinamas kiekvieną kartą, kai tik užbaigimo būseną suveikia. Serverio būsenos mašina parodo, kada serverio transakcija yra patvirtinimo būsenoje ir ne vienintelis įvykis I laikmačio iššaukimas įmanomas. Tačiau įmanoma gauti retransliuojamą *INVITE* ar *ACK* užklausą serverio pusėje, kai transporto terpė yra nepatikima (Hong, 2011). Būseną yra patvirtinama (confirmed), kai *INVITE* ar *ACK* yra gauta serverio transakcijoje, transakcija lieka toje pačioje būsenoje ir *INVITE* ar *ACK* užklausa yra ištrinama (Huang, 2006).



3.1 pav. Sesijos inicijavimo protokolo imitacinis modelis CPN tinklais

3.5. Sesijos inicijavimo protokolo laiko sąryšių modeliavimas taikant CPN

Kliento transakcijoje yra trys laiko žymekliai: A, B ir D. Laikmatyje B priskiriama kliento transakcijos laukimo maksimalus laikas laikinam ar galutiniam atsakui iš serverio pusės, esant skambinimo būsenoje. B laikmatis naudojamas nepriklausomai kokoje transporto terpėje transakcija yra atliekama.

Laikmatis A yra naudojamas tik kai terpė nepatikima kontroliuoti *INVITE* užklausų retransliacijas. Laikmatis D svarbus kai terpė yra nepatikima, kadangi jo vertė yra lygi nuliui, patikimam transportui ir 32 sekundes nepatikimam transportui. Nuskaitinėjant visas galimas įvykių sekas naudojant *CPN*, pirmiausia reikia aiškiai apibrėžti sąryšius tarp dviejų laiko žymeklių. Pagal RFC pradinė laiko žymeklio A vertė yra T_1 , kurio numatytoji vertė yra 500ms, lygi žinučių apsikeitimo laikui tarp kliento ir serverio.

Kiekvieną kartą laikmačiui A iššaukiant, jo atstatomoji vertė tampa dvigubai didesnė nei buvo iki šio momento. Laikmačio B vertė yra $64 * T_1$, galima pastebėti, kad prieš laikmačiui B iššaukiant, laikmatis A gali atsirasti 6 kartus (mažiau kaip 6 kartai, jei atsakas yra gautas ar įvyksta transporto klaidos prieš iššaukiant laikmačiui B, atitinkamai laiko intervaluose T_1 , $2 * T_1$, $4 * T_1$, $8 * T_1$, $16 * T_1$, $32 * T_1$ (pastaba, kad intervalų suma yra $63 * T_1$, šiek tiek mažiau negu $64 * T_1$, tai yra laiko žymeklis, kuris gali būti iššaukiamas ne daugiau kaip 6 kartus, kol laiko žymeklio B galiojimas baigiasi. Kiekvieną kartą, kai laiko žymeklis A yra iššaukiamas, *INVITE* užklausa yra retransliuojama. Kada *kliento transakcija* sukurta, pirminė *INVITE* užklausa retransliuojama ne daugiau kaip 7 kartus, kurios gali būti siunčiamos *kliento transakcijai* prieš iššaukiant laikmačiui B.

Serverio transakcija taip pat turi tris laiko žymeklius: G, H ir I (žr. 3.2 pav.) Laiko žymeklis H yra naudojamas abiejose patikimose ir nepatikimose aplinkose, įrašant maksimalų laiką, t.y laikas kiek serverio transakcija galėtų laukti esant sėkmingai įvykdytai būsenai iki ACK užklaustos gavimo. Laiko žymeklis G yra naudojamas nepatikimam transportui kontroliuoti intervalo 300-699 atsakų retransliavimus.

Kai sėkmingai įvykdyta būsenai, laikmatis H yra nustatomas į $64 * T_1$ ir laikmatis G yra nustatytas į T_1 . Kai laikmatis G iššaukiamas i -jame $i \geq 2$ laike, jis yra atstatomas į minimalią reikšmę $2^{i-1} * T_1$ ir T_2 ($T_2 = 8 * T_1$). Dėl tos priežasties, kol laikmačio H galiojimas baigsis, laikmatis G gali inicijuoti iki 10 kartų kvietimus intervaluose, T_1 , $2 * T_1$, $4 * T_1$, $8 * T_1$, $8 * T_1$, $8 * T_1$, $8 * T_1$, $8 * T_1$, $8 * T_1$, $8 * T_1$, $8 * T_1$ atitinkamai. Jei *ACK* yra gaunama ar transportavimo klaida įvyksta

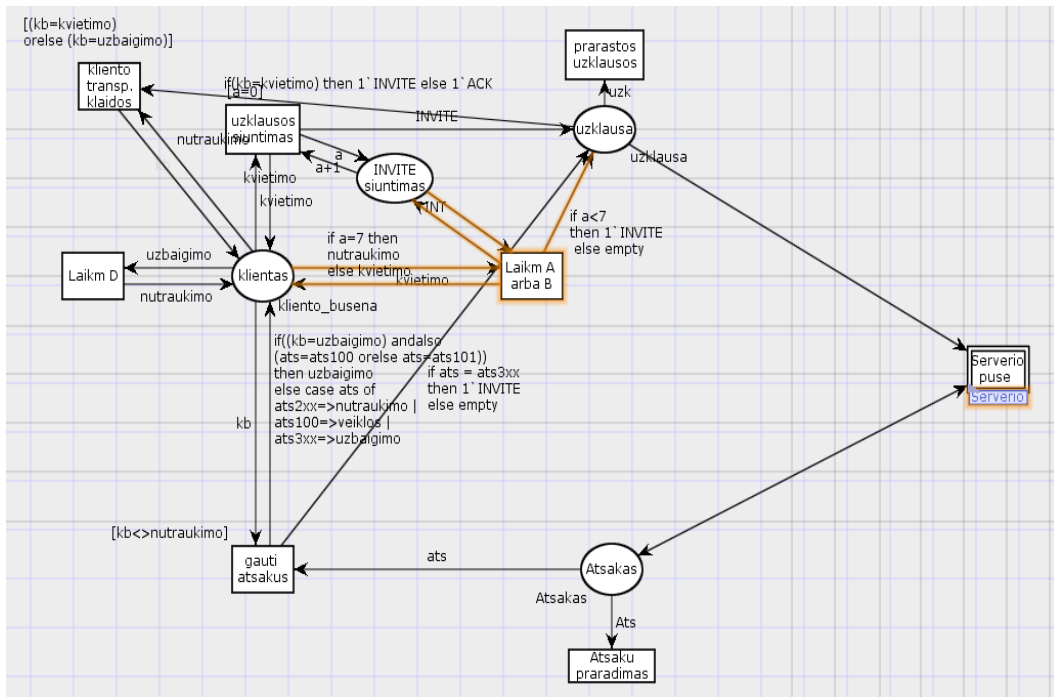
prieš iššaukiant laikmatį H, laiko žymeklis G iššaukiamas mažiau nei 10 kartų. Laiko žymeklis I yra nustatytas iššaukti iškart po nulio sekundžių patikimoje transporto sluoksnyje ir po 5 sekundžių nepatikimajame sluoksnyje. Norėdami išskirti serverio transakcijos būsenas iš *kliento transakcijos* būsenų su tuo pačiu vardu, pridedame S prie serverio transakcijos būsenų vardo (išskyrus *proceedingT* būseną). Pavyzdžiui, *TyrimasS* atvaizduoja serverio transakciją *Tyrimo* būsenoje, kol tyrimas atvaizduoja kliento transakcijos tyrimo būseną. SIP atsakymų žinutės (lentelė 1) pavadintos taip: r100 vaizduoja *Trying* (100) atsakymą; r1xx yra laikini atsakymai su statusų kodais tarp 101 ir 199; r2xx 2xx atsakams; ir r3xx 300-699 atsakams.

3.6. Sesijos inicijavimo protokolo kliento pusės transakcijų sekos aprašymas spalvotaisiais Petri tinklais

Kairiojoje 3.1 pav. CPN modelio pusėje yra „*Kliento*“ ir „*INVITE Siuntimas*“ pozicijos, sujungtos transakcija. Kliento pusės transakcijos modelių būsenų mašina (pateikta 3.2 pav.).

Spalvotų Petri tinklų pozicija „*Klientas*“ yra aprašoma spalvų rinkiniu *KlientoBusena*, modeliuojant *INVITE* kliento transakcijos būsenas. *KlientoBusena* spalvų rinkinys aprašomas taip: *colset KlientoBusena = with skambinama | tyrimas | sėkmingas | nutrauktas*. Pradinis žymuo yra *skambinama*. Žymuo „*INVITE Siuntimas*“ yra įvestas spalvų rinkinys *INT (colset INT = int with 0..11;)* ir naudojamas suskaičiuoti *INVITE* užklausų skaičių, kurios buvo perduotos ir retransliuotos (t.y. pakartotinai perduotos). Penki perėjimai yra sujungti su „*Klientas*“ pozicija. 3.1 pav. matyti, kad „*Užklausų Siuntimas*“ paima transakcijos *INVITE* užklausą į SIP transporto sluoksnį perdavimui (t.y. įdedamas *INVITE* žymuo į Request place (vietą)). *INVITE* žyma bus matoma tik tada, kai „*Klientas*“ *skambinama* būsenoje ir „*INVITE Sent*“ turi savyje sveiką skaičių 0 (kitos ne *INVITE* užklausos buvo išsiųstos). „*Atsakų gavimas*“ parodo, kaip kliento transakcija gauna atsakymus ir siunčia *ACK* žinutę. Jis yra įgalintas, kai atsakas yra gautas ir „*Klientas*“ nėra *nutrauktas* (*nutrauktas* būsenoje). Jei kliento transakcija gauna atsakymą intervale 300-699 (r3xx), *ACK* bus

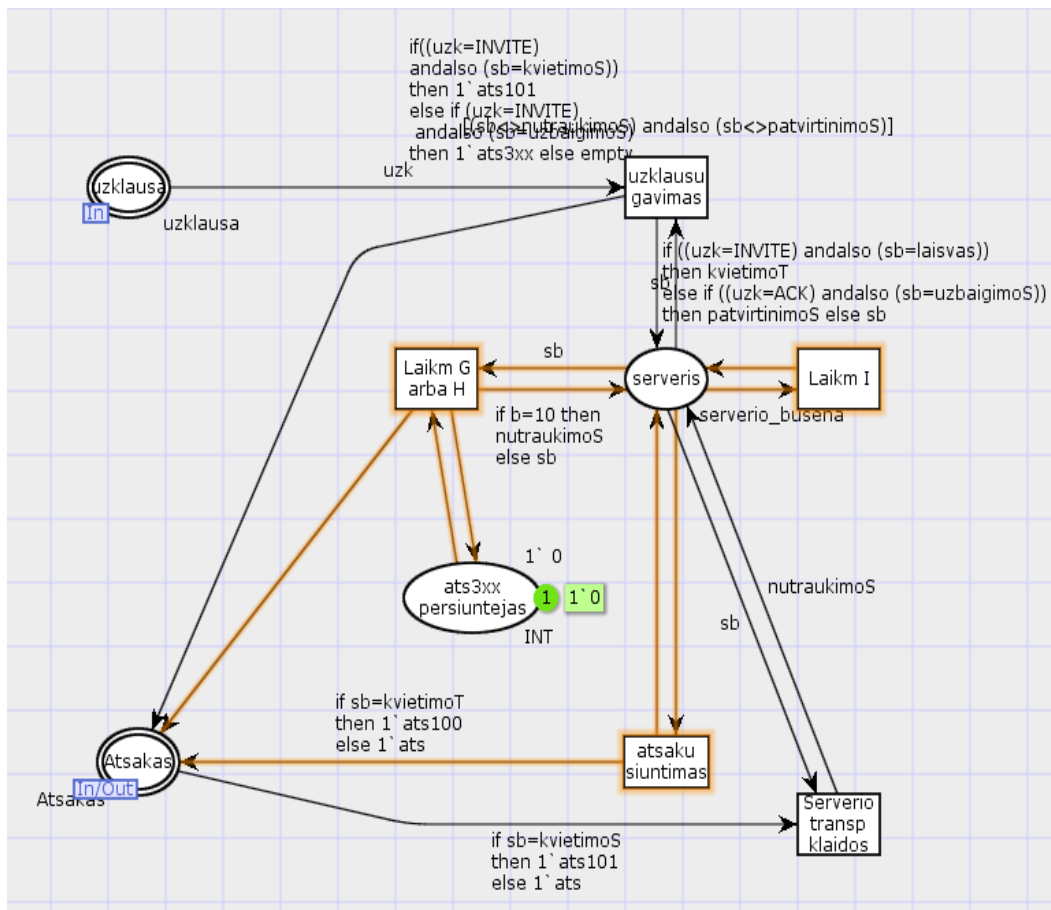
perduodama į SIP transporto sluoksnį (žr. 3.1 pav. nuo „užklausų gavimas“ iki vieta (*place*) Užklausa) ir „Klientas“ keičiasi į sėkmingai įvykdytą (*sėkmingas*) būseną. Kreipiamasi iš „Atsakų gavimas“ į „Kliento“ vietą, jei gaunamas atsakymas yra r100, r101 ar r2xx, ACK nėra siunčiamas, kai atsakymas yra r100 arba r101, Kliento vieta pasikeičia į *tyrimas* būseną; ir kai atsakymas yra r2xx, „Klientas“ būseną pasikeičia į *nutrauktas*. Įvedame du laikus A ir B ir modeliuojama taip: laiko žymė A negali pradėti tol, kol INVITE užklausa nėra išsiųsta. Leidžiama „INVITE siuntimas“ būti laiko A ir B įvesties vieta (*places*) ir transakcija įjungta tik tada, kai „INVITE siuntimas“ yra sveikas skaičius didesnis arba lygus 1 ir „Klientas“ yra *skambinama* būsenoje. Pradinis „INVITE siuntimas“ žymuo yra 0. Kai „užklausos siuntimas“ arba laikai A arba B įvyksta (INVITE užklausa yra siunčiama), sveikas skaičius padidinamas vienetu. Laikas B nesibaigia tol, kol laikmatis A nesibaigia 6 kartus arba 7 INVITE užklausos nebuvo išsiustos. Taigi lanko (*arc*) įrašas iš „INVITE siuntimas“ į laiko A arba B tikrina kintamojo *a* reikšmę. Kada *a* reikšmė yra $a < 7$, laikmatis A suveikia, todėl laikmačių A arba B suveikimai nepakeičia „Kliento“ žymėjimų, bet įdeda INVITE į „užklausa“ (*place*) vietą. Kai $a = 7$ perėjimo modelis iššaukia laikmatį B, kuris keičia „Kliento“ žymenis iš *skambinama* į *nutrauktas* būseną, bet neįdeda nieko į „užklausa“ vietą. Perėjimo laikmatis D yra įgalintas, kai yra *sėkmingas* būseną „Kliente“ ir jo pasirodymas keičia „Klientą“ į terminuotą būseną, modeliuojamas laikmačio D suveikimas. Perėjimas „kliento transportavimo klaida“ pasirodo, kai „Klientas“ yra *sėkmingas* būsenoje. Jo pasirodymas taip pat keičia „Klientas“ būseną į *nutraukimo*. Kai klaida yra pranešama SIP transporto sluoksnyje, ACK, kuri turi būti perduota iš transakcijos sluoksnio, nebebus siunčiama į serverio pusę, tada „kliento transporto klaida“ įvyksta. ACK, kuri įdedama į „užklausa“ vietą, yra nutraukimo būsenoje, tai matoma lanko iš „Kliento transporto klaidos“ į „užklausa“ užrašė. Transporto klaida gali atsirasti, kai kliento transakcija yra *skambinama* būsenoje, taigi „kliento transporto klaida“ yra įgalinta taip pat, kai „klientas“ yra *skambinama* būsenoje.



3.2 pav. SIP klientinės pusės modelio CPN tinklas

3.7. SIP serverio transakcijų modeliavimas

SIP serverio transakcijos modelis CPN tinklais pateiktas 3.3 pav. CPN dešinėje modelio pusėje yra serverio pozicijos vieta, r3xx atvaizdavimas ir 5 perėjimai sujungti su „Serveriu“. Serverio vietos spalvų rinkiniu *STATESERVER* modeliuojamos serverio transakcijos būsenos. Modelyje yra nauja, *KvietimoT* būseną, kuri įdėta į serverio transakciją. Kadangi transakcija yra sukurta ir įeina į *KvietimoT* būseną, tik kai transakcijos vartotojas gauna INVITE užklausą iš kliento pusės per SIP transportavimą tiesiogiai. Pradiniai „Serverio“ vietos žymėjimai negali būti *kvietimo* būseną. Tačiau vieta neturi pradinių žymenų, todėl pradinius žymenis žymėsime kaip *laisvas* būseną, modeliuojamą taip, kad serverio pusė būtų pasiruošusi gauti INVITE užklausą iš serverio pusės. *Ats3xxpersiuntejas* pozicijoje yra INT tipo įrašai ir retransliacijos r3xx skaičių įrašai, suveikiant laikmačiui G.



3.3 pav. SIP serverio pusės imitacinis modelis CPN tinklais

3.8. Galimos sesijos inicijavimo protokolo skambučių būsenos

SIP INVITE kliento transakcija leidžia palaikyti funkcionalumą pasitelkus būsenų mašiną (pateikta 2.1 pav.). SIP RFC 3261 apibrėžia šias Kliento INVITE transakcijos būsenas šioje būsenų mašinoje.

Skambinimo būseną: Ši būseną rodo, kad Vartotojo Agentas inicijuoja naują kliento transakciją su INVITE užklausa. Jei naudojamas nepatikimas transportavimo sluoksnius, kliento transakcija inicijuoja laikmatį A. Laikmatis A kontroliuoja užklausų retransliavimą. Jei laikmatis A suveikia, tada kliento transakcija lieka toje pat būsenoje, o INVITE užklausa turi būti retransliuojama. Laikmatis B taip pat startuoja tokioje pat būsenoje. Jei laikmatis B suveikia, tada kliento transakcijos pereina į nutraukimo būseną.

Tyrimo būseną: ši būseną nurodo, kad kliento transakcija gavo tarpinį atsakymą, kol yra „calling“ būsenoje. Šioje būsenoje kliento transakcija neturi retransliuotų užklausų.

Užbaigimo būseną: ši būseną nurodo, kad kliento transakcija gavo 300-699 ribose atsakymą. Kliento transakcija turėtų pradėti laikmatį D būsenoje. Ši būseną nurodo, kad prasideda laikmačio D arba laikmačio B veikimas. Kliento transakcija taip pat įveda šią būseną, jei gauna 2xx atsakymą, kol yra tyrimas būsenoje. Kliento transakcija turi būti sunaikinama akimirksniu, jei patenka į šią būseną. ACK yra siunčiamas po 2xx užklauso atsako ir reiškia, kad SIP skambutis yra „vykdomas“ (UP).

3.9. Laikmačio B verčių keitimasis

Jei kliento transakcija yra *skambinama* būsenoje ir laikmatis B suveikia, tada jis tyliai perleidžia kliento transakciją, kai kviečiančiosios pusės INVITE transakcija pereina į nutraukimo būseną. Tačiau kitam vartotojui, kuriam skambinama, transakcija vis dar bus *tyrimo* būsenoje ir jam skambės telefonas.

Laikmatis B kontroliuoja transakcijos užlaikymo mechanizmą. Kliento transakcija privalo prasidėti ne vėliau kaip $64T_1$. Visos transakcijų laikmačių skalės matuojamos T_1 ir koreguojamos T_1 keičiant transakcijos laiką. T_1 į abi puses įvertinamas laikas.

Kiekvieną kartą siunčiant užklauso (*request*), siunčiami šie duomenys: CallID, FromNodeID, ToNodeID, SIPMethod, SIPMethod, MsgSize, Expire.

CPN šių parametrų pokyčių aprašas atrodytų taip:

```
colset SReq = product CallID * FromNodeID * ToNodeID * SIPMethod *  
SIPBody * MsgSize * Expire;
```

```
colset SResp = product CallID * FromNodeID * ToNodeID * SIPRespCode *  
SIPBody * MsgSize * Expire;
```

```

▼Declarations
▶Standard priorities
▼Standard declarations
▼colset UNIT = unit;
▶colset BOOL
▶colset STRING
▼colset INT = int with 0..11;
▼colset uzklausa = with INVITE | ACK;
▼colset kliento_busena = with kvietimo | veiklos | uzbaigimo | nutraukimo;
▼colset serverio_busena = with laisvas | kvietimoT | kvietimoS | patvirtinimoS | uzbaigimoS | nutraukimoS ;
▼colset Atsakas = with ats100 | ats101 | ats2xx | ats3xx;
▼colset atsakai = subset Atsakas with [ats101, ats2xx, ats3xx];
▼var kb : kliento_busena;
▼var a,b :INT;
▼var uzk: uzklausa;
▼var sb :serverio_busena;
▼var ats:atsakai;
▼var Ats:Atsakas;
    
```

Simboliai ++ ir ‘ yra operatoriai, naudotini konstruojant skirtingus rinkinius, sudaryti iš šių spalvų.

3.10. Judančių objektų komunikacijos imitavimo privalumai taikant spalvotuosius Petri tinklus

Yra galimybė stebėti judančio objekto būseną mobiliosiomis technologijomis. Sensorinės įrangos sensorių priemonių atitinkamos parametrų reikšmės gali būti perduodamos į nutolusius serverius, kurie šiuos duomenis susistemina, informuoja ir nusako tolimesnius veiksmus. Norėdami tinkamai valdyti nutolusius objektus turime susieti kontekstinę informaciją su gaunamais duomenimis, juos diagnozuoti ir įvertinti jų būklę. Kadangi nutolę objektai nėra iš anksto nuspėjami, turime numatyti visas įmanomas situacijas. Tikrinti scenarijus programuojant sugaišina daug žmogiškųjų resursų. Siūloma pirmiausia suprojektuoti sistemą su spalvotais Petri tinklais ir išbandyti visus galimus scenarijus. Spalvotaisiais Petri tinklais pakankamai greitai projektuojamos sistemos, kurios, yra lengvai suprantamos grafiškai atvaizduotos sistemos.

Tikslas – sumodeliuoti imitacinę sistemą, kuri niekuo nesiskirtų nuo realiai veikiančios sistemos, taikant spalvotųjų Petri tinklų projektavimo metodologiją. Sistemos projektavimo ir modeliavimo žingsniuose stengiamasi įvertinti tokius kriterijus, kaip: našumas, lankstumas, kaina, patikimumas.

Spalvotieji Petri tinklai yra grindžiami intuityviu grafiniu atvaizdavimu; yra įvykdomi; gali būti konstruojami hierarchiniai modeliai; yra galimybė modeliuoti įvairių sistemos procesų sugaištą laiką.

Jais galima nagrinėti sistemą ypač detaliai, nes kiekvienas sistemos žingsnis aprašomas atskirai. Paslaugų sistemoje kiekviename žingsnyje yra naudojama Petri tinklų architektūra. Kiekviena sistemos posistemė turi būti suskaidoma į komponentus – operacijas ir būsenas. CPN modeliuoja sistemos būsenas. Ryšiai reikalingi keičiant vieną būseną kita ir atliekant valdančios informacijos ar materialiujų srautų judėjimą. Sistemos būseną yra nustatoma „būsenos žymėmis“ ir yra fiksuojama tam tikroje pozicijoje. Būsenos žymės dar vadinamos pozicijomis.

Pozicijos atitinka objektus (resursus, signalus, duomenis) modeliuojamoje sistemoje. Dažnai norima atvaizduoti šių objektų atributus (numerį, pavadinimą, kiekį ir pan.). Tai nėra paprasta padaryti naudojant klasikinio Petri tinklo žymes, todėl įvedamos spalvotos žymės. CPN kiekvienos žymės tipą ar reikšmę nurodo jos spalva. Perėjimai aprašo gaminamų žymių spalvas, remiantis sunaudotų žymių spalvomis (aprašo ryšį tarp įėjimo žymių ir išėjimo žymių). Taip pat galima aprašyti „prieš sąlygas“, kurios atsižvelgia į sunaudojamų žymių spalvas. Spalvoti Petri tinklai suteikia galimybę pažymėti žymių tipus skirtingomis spalvomis, kurios gali atspindėti sutartines duomenų reikšmes. Kiekviena pozicija gali būti susieta su tam tikru tipu, kuris apsprendžia duomenų tipą, kurį ji gali saugoti.

CPN leidžiama atlikti interaktyviąją imitaciją, kurios metu rezultatai pateikiami tiesiogiai diagramoje. Imitacijoje galima lengvai sumodeliuoti didelę ir sudėtingą sistemą, kurioje galima stebėti ir, reikalui esant, keisti judančių žetonų „pernešamą“ informaciją. Taip pat leidžiama atlikti sistemos būsenos analizę, t. y. konfliktuojančias būsenas, konkurencinius veiksmus, sinchronizacijos problemas, pasikartojančių būsenų ir bendrų resursų panaudojimo galimybes.

CPN yra įrankis, kuriuo patvirtiname diskrečius sistemos įvykius, t. y., jie yra tiriami ir modeliuojami. CPN naudojama analizuoti ir gauti naudingos

informacijos iš struktūrų ir reikalingos dinaminėms modeliuojamoms sistemoms. Grafinės CPN savybės patikslina taikymą ir modeliuojamos sistemos vizualizaciją. Be to, sinchroniniai ir asinchroniniai įvykiai pateikia savo prioritetus sąryšiuose ir struktūrinius prisitaikymo efektus. Protokole naudojamos skaičių sekos, patvirtinimai ir retransliavimas užtikrina, kad duomenų paketai būtų pristatomi tik kartą ir teisinga tvarka. Protokolas taiko strategiją sustabdyti-ir-laukti, t. y., tie patys duomenų paketai yra pakartotinai retransliuojami, kol priimantysis galas patvirtina.

3.11. Mobilųjų objektų komunikacijos ir būsenos modeliavimas bei valdymas taikant spalvotuosius Petri tinklus

Geriausias galimybes stebėti judančius objektus ir analizuoti jų būsenas šiuo metu suteikia mobiliosios technologijos. Naudojami metodai ir programinė įranga belaidėse sistemose leidžia keistis daugumai įmanomų duomenų formatų (pvz., tekstiniais, balsiniais ar vaizdiniais), suteikia informaciją apie objekto būseną geografinėje vietovėje realiuoju laiku. Reikiama informacija gali būti suteikiama iš sensorių. Tačiau mobilaus įrenginio kontekstinė informacija ir jo aplinka reikalauja papildomų įvertinimo priemonių. Esant poreikiui informacija yra siunčiama į nutolusius serverius ir atlikus tam tikrus algoritmus bei skaičiavimus galima gauti tikslesnius duomenis. Situacijos nustatymui turime suprojektuoti ir įdiegti situacijos nustatymo programinę ir techninę įrangą, kuri galėtų išsiųsti duomenis, perspėjimo signalus ar priminimą susidariusiai atitinkamai situacijai vertinti. Darbe pateikiame metodiką, kaip taikyti imitacinio modeliavimo priemones – spalvotuosius Petri tinklus (CPN) – nutolusių mobilių objektų stebėjimui ir būsenų fiksavimui, situacijos modeliavimui bei ryšio priemonių valdymui. Mobilieji įrenginiai per sensorines priemones leidžia fiksuoti duomenis apie stebimą objektą bet kuriuo metu, bet kurioje vietoje. Tačiau šios informacijos analizei reikalingi sudėtingesni žinių išskyrimo ir didelių duomenų saugyklų metodai. Modeliai, sukurti CPN formaliomis priemonėmis, leido numatyti iš anksto sumodeliuotas situacijas skirtingais detalizavimo lygmenimis

ir tinkamai fiksuoti bei analizuoti stebimą informaciją apie judantį objektą, pasitelkiant tinkamas taisykles, kurios ir sudarė žinių bazės pagrindą.

Mobilusis įrenginys aptinka reikiamus duomenis per išorines ar vidines fizines įrangos aplinkas (sensorius). Įrenginio komponentai bendrauja su vidine ar išorine fizine aplinka ir turi tų sensorių parametrus. Vertinimas vyksta pasitelkus žinių bazę, kurioje aprašomos vertinimui reikiamos taisyklės, išskviečiamos pagal situacijos pradinius meta-duomenis.

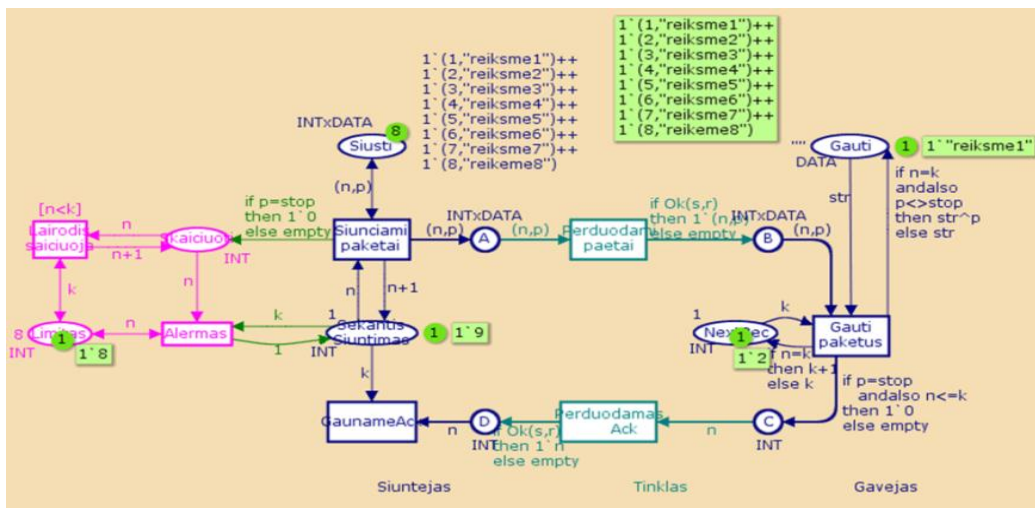
Technologinė įranga, skaitanti duomenis iš sensorių, yra pagrindinis komponentas šių duomenų surinkimui ir duomenų saugyklos. Sensorinės įrangos sensorių atitinkamų parametrų reikšmės gali būti perduodamos ir nutolusius serverius, kuriuose šie duomenys yra sisteminami ir taikomi analizei bei informacijai dėl tolimesnių objekto valdymo veiksmų. Norėdami tinkamai valdyti nutolusius objektus, turime susieti kontekstinę informaciją su gaunamais duomenimis, juos diagnozuoti įvertinant būklę. Kadangi nutolę objektai nėra iš anksto nuspėjami, mes turime sumodeliuoti kiek galima daugiau įmanomų scenarijų situacijų aprašymui.

Programuojant scenarijus sudėtingos sąveikos sistemose, svarbu surasti tinkamus būdus tokių sistemų modelio sukūrimui, kad būtų galima tinkamai juos taikyti mobilių įrenginių komunikavimo ir stebėsenos procesų valdymui. Tokių sistemų aprašymui ir imitaciniam modeliavimui bei realiai kontrolei atlikti taikomi spalvotieji Petri tinklai leido atvaizduoti projektuojamos sudėtingos sistemos skirtingus detalumo lygmenis, lengvai suprasti ir grafiškai perteikti atvaizduojamas sistemos savybes ir procesų dinamiką.

Imitaciniai modeliai įgalina lengviau atvaizduoti procesus, kurie tinkamai atspindėtų veikiančių objektų elgseną. Šie tinklai yra Petri tinklų formalizmo praplėtimas, leidžiantis atvaizduoti realybėje vykstančių procesų parametrų pokyčių dinamiką (Dzemydienė, Dzindzalieta, 2010), ją stebėti ir valdyti. Sukurti modeliai turėtų būti taikomi belaidžių įrenginių infrastruktūroje integruojant juos ir bendrą judančio objekto stebėsenos ir būsenos nustatymo bei atpažinimo sistemą (Dzemydienė ir kt., 2010) kaip vieną iš žinių bazės komponentų.

CPN yra įrankis, kuriuo patvirtiname diskrečius sistemos įvykius, t. y., tiriamus ir modeliuojamus procesus. Grafinės CPN savybės patikslina taikymų galimybes ir teikia tinkamas priemones modeliuojamos sistemos daugialygei vizualizacijai. Be to, sinchroniniai ir asinchroniniai įvykiai pateikia savo prioritetus sąryšiuose ir struktūriniuose prisitaikymuose. Pagrindinis skirtumas tarp CPN ir klasikinių Petri tinklų yra tas, kad CPN elementai yra atskiriami tam tikrų spalvų priemonėmis, kurios netaikomos tradiciniuose Petri tinkluose (Billington et al. 2004).

Tiriant belaidžių įrenginių komunikavimo protokolų sistemos aprašymą, nagrinėjamos sesijos inicijavimo protokolo (SIP) praplėtimo galimybės. Čia CPN teikiamos savybės integruojamos į SIP aprašą ir taikomi tokie parametrai kaip skaičių sekos, patvirtinimai ir retransliavimas, kurie užtikrina, kad duomenų apdorojimo paketai būtų pristatomi tik kartą (nedubliuojami) ir vykdomi teisinga tvarka 3.4 pav. Protokolas taiko strategiją „sustabdyti-ir-laukti“, t. y., tie patys duomenų paketai yra pakartotinai retransliuojami, kol priimantysis tikslinis objektas patvirtina gavimą pagal veikimo taisyklės.

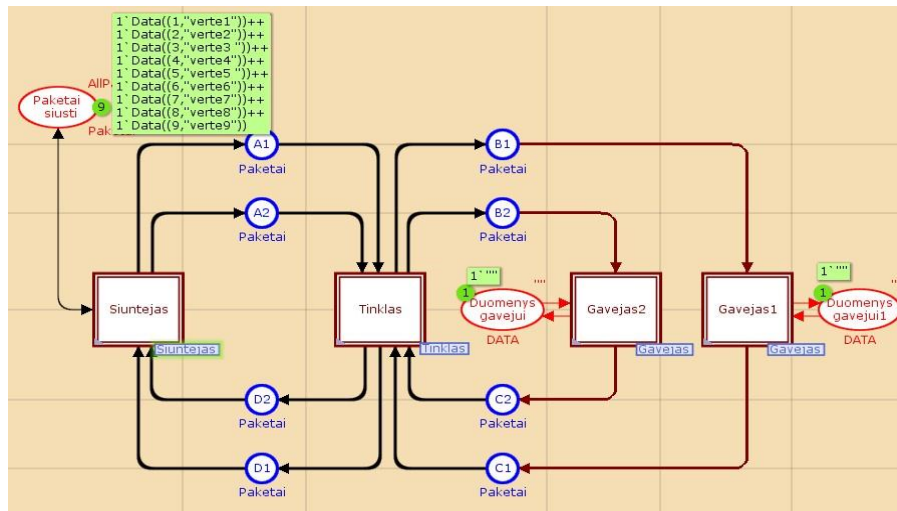


3.4 pav. CPN schema, perduodamų duomenų paketus

3.12. SIP programavimo sąsaja mobiliesiems įrenginiams ir sesijų valdymas

Sesijos inicijavimo protokolo programavimo sąsaja mobiliesiems įrenginiams ir sesijų valdymas yra svarbiausias uždavinys judančių objektų stebėsenoje.

Norint pritaikyti sesijos inicijavimo protokolo (SIP) technologiją mobiliesiems įrenginiams, reikia pasinaudoti specifiniu javax.microedition.sip paketu, kuris užtikrina susijungimo sesijas tarp SIP klientų (Huang, Lee, 2006). SIP klientai naudoja TCP ar UDP (dažniausiai 5060 portą), kad prisijungtų prie SIP tarnybinės stoties ar prie kitų galinių sistemų. Bendra SIP veikimo struktūra, atvaizduojama CPN modeliavimo priemonėmis, pateikiama 3.5 pav.



3.5 pav. Bendra SIP veikimo struktūra, atvaizduojama CPN modeliavimu

Šioje sistemoje svarbu inicijuoti ir užbaigti kliento ir serverio ryšius, valdyti duomenų siuntimo srautus. Šiam uždaviniui išspręsti tenka apibrėžti javax.microedition.Connection paketo klases, kurios būtinos, kad galėtume panaudoti SIP technologiją. Mobilieji įrenginiai turi teikti savo fizinius adresus SIP agentui, kad stebėtojas galėtų bet kuriuo metu juos rasti. Norint užmegzti sesiją su kitu įrenginiu, siunčiama Invite žinutė, kuri realizuoja įrenginių susijungimą. Dviejų ar daugiau įrenginių sujungimas pradedamas išsiunčiant SIP Invite žinutę prieš pradedant seansą. Per seansą inicijuojama užklausa, kuri leidžia virtualiai susijungti dviem ar daugiau įrenginių ir pasikeisti duomenimis. Pirmiausia SIP Invite užklausa atsako mums statusu, kurio kodas yra 180, reiškiančio, kad kitą klientą pasiekė informacija apie skambutį. Apie sėkmingai pateiktą žinutę informuoja kodas 200. Norint užbaigti SIP seansą tarp įrenginių, siunčiama SIP Bye žinutė, kuri nutraukia sesijos darbą.

3.13. Trečiojo skyriaus išvados

Šiame skyriuje aprašyta metodika kaip pritaikyti spalvotuosius Petri tinklus SIP funkcinių savybių aprašymui, modeliavimui ir analizei. Spalvotieji Petri tinklai leido kiekviename sistemos žingsnyje nustatyti reikiamus realios situacijos vykdymo parametrus, aprašyti tinkamus algoritmus ir vykdomų paslaugų atlikimo scenarijus.

Skyriuje apžvelgiamos paslaugų atlikimo galimybės, kiekvienoje sistemos posistemėje. Sistema suskaidyta į komponentus (pvz., kliento komunikacijos posistemę, serverio komunikacijos posistemę) CPN modelyje leido ištirti atliekamas operacijas ir stebėti vykdomas imitaciniame modelyje sistemos būsenas.

Spalvotųjų Petri tinklų priemonėmis sukurti modeliai leido išanalizuoti ryšių kanalais perduodamos informacijos struktūrą, kuri reikalinga keičiant vieną būseną kita, bei matyti kaip jais atliekami valdančios informacijos perdavimai ir informacijos srautų judėjimai.

4. SPRENDIMŲ PARAMOS SISTEMOS ARCHITEKTŪRA PAVOJINGŲ KROVINIŲ STEBĖSENAI IR BŪKLĖS VERTINIMUI TAIKANT MOBILIĄSIAS TECHNOLOGIJAS

Skyriuje pateikiama sprendimo paramos sistemos (SPS) architektūra, kuri naudoja belaides technologijas stebint judančias transporto priemones. Aprašoma SPS architektūra su įterptinėmis posistemėmis judančių objektų stebėsenai, lokalizacijai ir būklės vertinimui.

4.1. Sprendimų paramos sistemos judančių objektų stebėsenai samprata

Sprendimų paramos sistema – tai kompiuterizuota sistema, padedanti sprendimų priėmėjui suformuoti uždavinius, juos išspręsti ir priimti sprendimus, naudojant informacines technologijas, duomenis, modelius ir žinias. Sprendimo paramos sistema – interaktyvi informacinė sistema, skirta formuoti reikalingą informaciją sprendiniams priimti ir padedanti sistemos vartotojams spręsti struktūrizuotas ir struktūnestruktūrizuotas problemas naudojant duomenis ir modelius.

SPS galima suvokti kaip įvairių šaltinių duomenų kaupimo ir tų duomenų apdorojimo sistemą, kurios pagalba sistemos vartotojai gali priimti specifines, nestruktūrizuotas ar iš dalies struktūrizuotas problemas. Nagrinėjant SPS galima suprasti, kad ji ne tik padeda priimti sprendimus, bet parenka priimtinausią variantą iš jai pateikiamų alternatyvų ar jos formuojamų alternatyvų (Dzemydienė, 2006, Janakiraman, 2008).

Kuriant SPS sistemas turi būti įvykdomi šie reikalavimai: sistemos ir vartotojų sąveikos užtikrinimas; modelių ir duomenų prieigos ir jų paieškos sujungimas į visumą; atsižvelgimas į vartotojų nuostatus bei aplinkos pokyčius.

Sukurta SPS sistema turi vykdyti šias funkcijas: sąsaja vartotojui užtikrinti; atlikti problemos atpažinimą; pateikti alternatyvius sprendimus problemos sprendimui. SPS funkcijos yra šios: 1) sąveika su sprendėju; 2) problemų atpažinimas; 3) problemos sprendimų siūlymas; 4) pasiūlyto sprendimo pagrindimas.

SPS sistema teikia informaciją, kurios reikia norint sudaryti alternatyvas, analizuoti, vertinti, priimti tinkamiausius sprendimus ir galimybę toliau plėtoti sistemą parenkant problemų sprendimo būdus (Kaklauskas, 2002).

Svarbu sukurti sprendimus palaikančią sistemą, kuri padėtų stebėti ir operatyviai vertinti transporto objektus, vežamų cheminių medžiagų būklę, kelio ruožų avaringumą. Ypatingą reikšmę įgyja transportavimo kelyje išsidėsčiusių teritorijų statusas ir padėtis. Šiam tikslui tenka ieškoti metodų, tinkamų vertinti pavojingų krovinių transportavimo procesus ir juos aprašyti SPS. Transportuojant krovinius, vežančius pavojingas, taršias medžiagas, išauga aplinkos taršos rizika, susijusi su judančių transporto objektų įvykių pasireiškimo tikimybinio padidėjimu ir atitinkamų pasekmių galimybėmis. Tikėtinos taršos rizika įgyja daugiafunkcinį charakterį ir yra sudėtingesnė nei rizikos vertinimas stacionarių objektų aplinkoje (Arpaia, 2005). Darnios aplinkos vystymo reikalavimai įpareigoja užtikrinti saugius transportavimo procesus ir ypač saugų pavojingų krovinių transportavimą skirtingose autotransporto priemonėse ir teritorijose.

Mobilios technologijos padeda nustatyti objekto buvimo vietos informaciją, leisti sieti ją su vietovės informacija, taikyti atitinkamus sensorių parodymus. Vietovės nustatymas grindžiamas priminimų paslauga ir veikia belaidžiuose įrenginiuose. Priminimai gali būti naudingi, kai kontekstinė informacija pateikiama tinkamu metu ir tinkamoje vietoje. Prietaisų sensoriai sąveikauja su fizine aplinka per sensorius (Peterkin, 2011, Bielskis, Dzemydienė, 2009). Pavojingų krovinių gabenimo kelias ilgas ir kiekviena akimirka keičiasi nuo daugelio faktorių, todėl pavojingų krovinių vežimui keliami ypatingi reikalavimai ne tik šalies, bet ir tarptautiniu mastu. Nėra vieningos sistemos, kuri koordinuotų pavojingus krovinius (Valente, 2009). Nėra sudarytos informavimo sistemos apie pavojingų krovinių būklę kiekvienu laiko momentu, kas leistų vairuotojui priimti reikiamas išvadas.

Per Lietuvą kasdien vežama daugybė tonų pavojingų krovinių, kurie gali išsilieti, sprogti ar kitaip paveikti žmonių sveikatą, turtą ar aplinką. Pagrindinės avarių priežastys gali būti šios: transportuojančių krovinių vežimo reikalavimų

nesilaikymas; nėra tinkamos kontrolės arba šios kontrolės nesilaikoma, arba kontrolė yra neveiksminga.

Teikiamų darbų tikslas yra teoriškai iširti ir parengti pasiūlymus, kaip efektyviai organizuoti pavojingų krovinių vežimą.

Rizikos kontrolė yra svarbi rizikos valdymui. Pavojingų medžiagų išmetimas į aplinką avarijos momentu yra rimta pasekmė. Pavojingų krovinių avaringumas priklauso nuo kelio charakteristikos ir transporto priemonės tipo. Tačiau kelių ir transporto priemonių savybes sunku pakeisti transportavimo procese.

Rizikos valdymas – tai struktūrizuotas, suderintas ir nenutrūkstamas procesas, padedantis nustatyti ir įvertinti galimybes ir pavojus, turinčius įtakos tikslams pasiekti, taip pat leidžiantis priimti sprendimus, atsižvelgiant į esamus veiksmus. Rizikos valdymas padeda pasiekti geresnių rezultatų ir iš esmės sumažina nuostolių patyrimo galimybę. Rizikos suvaldymas yra procesas, glaudžiai susijęs su pavojingo transporto duomenų nuskaitymu ir interpretavimu.

Rizikos analizė rodo, kad pavojingų medžiagų gabenimas yra sudėtingas procesas ir sukelia visai kitokius pavojus nei statiniai objektai.

Aplinkos tvarumas priklauso nuo saugaus transportavimo, ypač pavojingų krovinių saugaus gabenimo skirtingomis transportavimo rūšimis (Qiulin, 2010).

Siūlomi sprendimai, kaip projektuoti atitinkamas sąsajos struktūras: sprendimų paramos sistemų įterptinius komponentus, scenarijus, vykdančius paslaugų kontrolę judančio objekto konkrečios situacijos atpažinimui ir valdymui. Sprendimų paramos sistemai taikomi išskirstytų informacinių sistemų ir belaidžio ryšio sistemų sąveikos komponentai, t.y. programavimo moduliai, protokolai, sensorinių duomenų analizė.

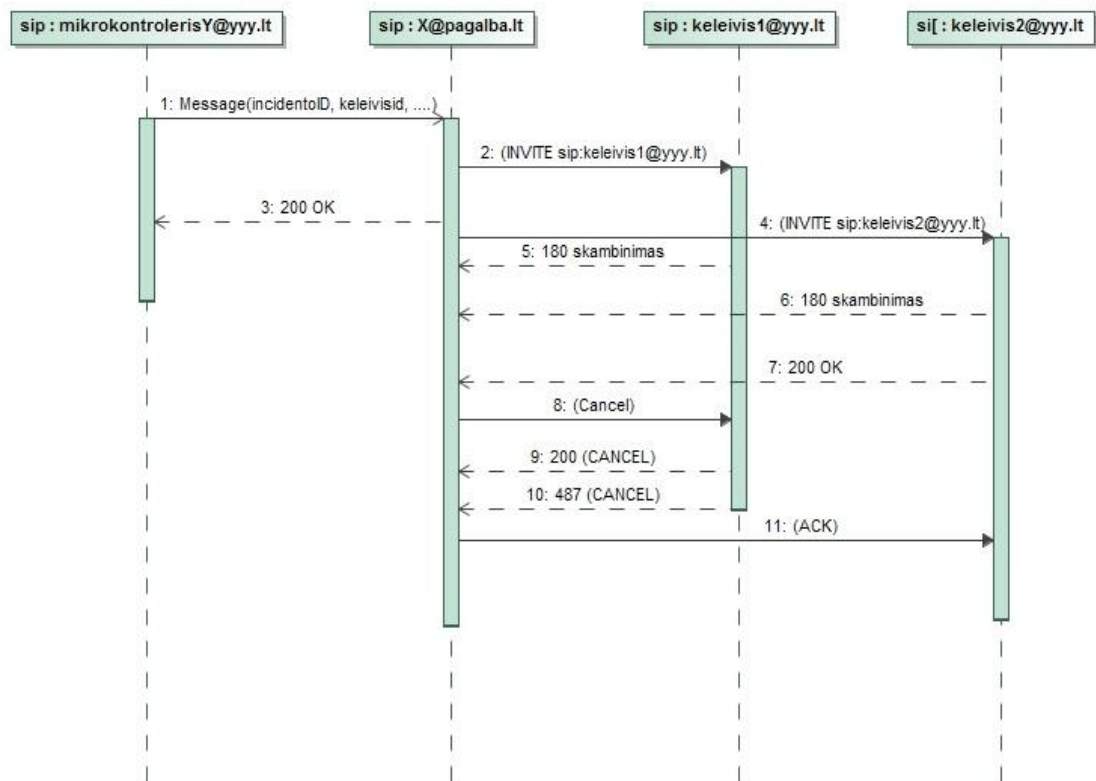
Pasitelkiant belaides technologijas judantiems objektams stebėti, susijusiems su pavojingų krovinių transportavimu, siūlome taikyti belaidžio ryšio protokolus, nustatant objekto geografines koordinatas, pateikiant įspėjimus apie judančio pavojingo transporto būseną ir galimų prognozių rezultatus panaudoti operatyviam objekto valdymui (Batarliene, Baublys 2007).

Pavojingų krovinių vežimas yra ne tik transporto, bet ir aplinkosaugos problema. Pavojingų krovinių vežimas kelia grėsmę žmonėms ir arti esantiems

objektams bei transportui. Nelaimingi atsitikimai susiję su pavojingais krovniais reikalauja daug dėmesio. Todėl daug dėmesio buvo skirta tyrimams, susijusiems su transporto priemonių modeliavimu priklausomai nuo susidariusios situacijos.

Įvykus incidentui, transporto priemonėje įmontuoti sensoriai automatiškai suaktyvins pagalbos iškvietimo sistemą, užmezgdamos ryšį su pagalbos tarnybomis. Suaktyvinta automobilyje įmontuota sistema siunčia pagalbos pranešimą ir užmezga balso ryšį (balso skambučiu), įskaitant pagrindinę informaciją apie avariją: avarijos laiką, vietą, automobilio kontekstinį aprašymą ir važiavimo kryptį (nustatytą remiantis tiksliais duomenimis, gautais per palydovines sistemas, bazinių stočių apskaičiuotą atstumą ar netoli esančių kitų transporto priemonių apskaičiuotą atstumą). Taip pat pagalbos iškvietimo sistemą galima aktyvuoti rankiniu būdu.

Automobilio vietą ir važiavimo kryptį galima matyti geografinės informacijos sistemoje gaunant pozicijos istoriją.



4.1 pav. Žinučių siuntimo lygiagrečio sujungimo sekų diagrama, pastebėjus sensorių duomenų nuokrypius

Sistema suprojektuota taip, kad įvykus incidentui, pagalba nebūtų siunčiama nereikalingai ar siunčiant skubos tvarka. Per sensorių sistemą galima bus girdėti, kas vyksta transporto priemonėje ir, esant galimybei, bus galima susisiekti su automobilyje esančiais keleiviais. Vienu metu bus inicijuojamas sujungimas su kitais automobilyje esančiais mobiliais įrenginiais, lygiagrečiai išsiunčiant užklausas ir bet kuriam užmezgus sėkmingą ryšį kitų įrenginių sujungimas nutraukiamas. Toks sujungimo būdas, kada vienu metu inicijuojamas sujungimas su keliais įrenginiais vadinamas lygiagrečiu sujungimu (4.1 pav.). Tai padės operatoriui įsitikinti, kokių pagalbos tarnybų reikia avarijos vietoje (greitosios pagalbos, gaisrininkų, policijos), ir greitai perduoti pranešimą apie avariją ir išsiųsti visą svarbią informaciją reikiamai pagalbos tarnybai. Norint padidinti efektyvumą, bus informuojami netoliese esantys eismo dalyviai, jog toje vietoje įvyko incidentas, kad kiti galėtų išvengti avarijų, taip pat bus nurodomi tolimesni veiksmai, kurie galėtų palengvinti kitų eismo dalyvių važiavimą. Taip kiti eismo dalyviai išvengs avarijų, padės atlaisvinti važiuojamąją kelio dalį ir sumažinti spūstis.

4.2. Sprendimo paramos sistemos pagrindinių komponentų architektūros aprašas

Skyriuje pateikiami pagrindiniai SPS projektavimo principai. SPS pagrindinių komponentų sąveikos schema pateikta 4.2 pav. SPS sistema sudaro keletas įterptinių posistemų. Gali būti įterpta realaus laiko posistemė, kuri gali būti įrengta transporto priemonėje. Tokia posistemė gali turėti savo skaičiavimo resursus, dirbant su stebimais duomenimis.

Būtiną sprendimų paramos modelio sukūrimas, šio modelio išsaugojimas ir to modelio pakartotinis panaudojimas. SPS – tai priemonių rinkinys, kuris konkrečiu atveju kreipiasi į aprašytus informacijos išteklius tam, kad gautų sensorių informaciją, ją apdorotų ir pateiktų vartotojams atitinkamu pavidalu.

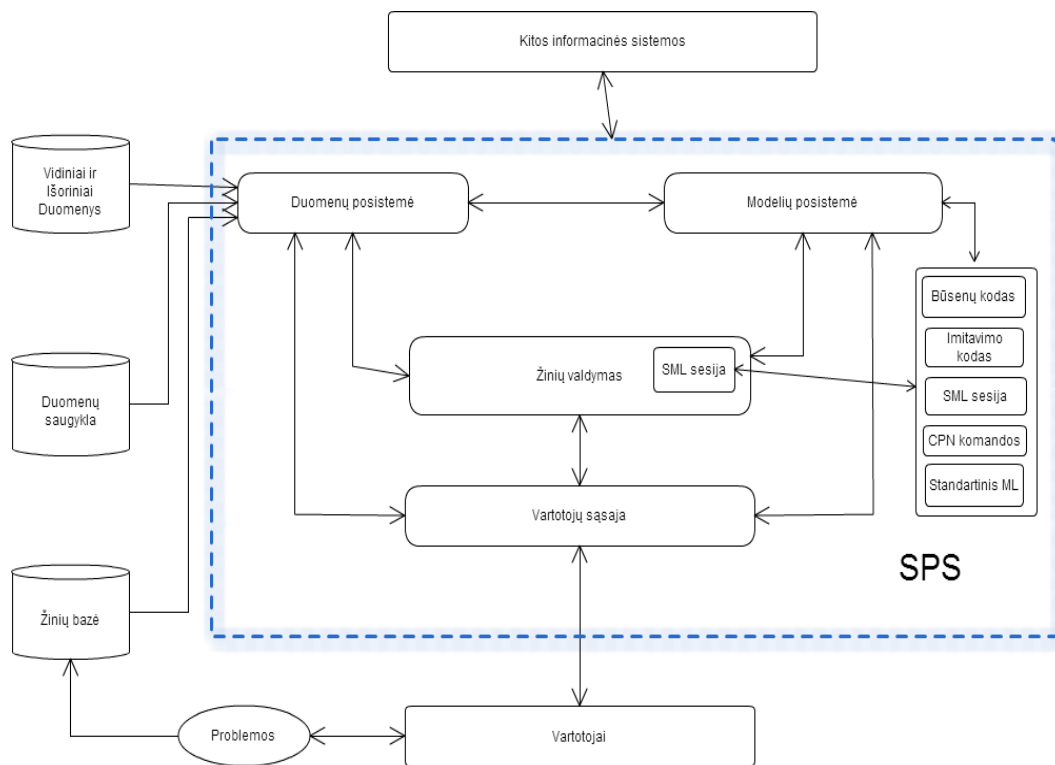
Projektuojama SPS sistema sudaryta iš pagrindinių posistemų:

- Duomenų apdorojimo posistemės, kurioje saugomi visi reikiami duomenys;

- Modelių valdymo posistemė, kurioje saugomi sukurti modeliai ir modelių sudedamosios dalys. Stebint pavojingus krovinius ir diagnozuojant jų būsenas reikalingi modeliai, kurie aprašytų kelio ruožų avaringumą, įvertintų avarių tikimybes, pateiktų pavojingų įvykių galimus scenarijus ir įvertintų galimos rizikos laipsnį;
- Sąsaja su judančiu objektu ir jų stebėsenos sensoriais. Ši sąsaja – svarbi sistemos dalis;
- Sąsaja su vartotoju (sprendimo priėmėju) yra sistemos dalis, nes sprendimai gali būti priimami bendraujant vartotojui su sistema.

SP sistemos bendroji architektūra pateikta 4.2. pav. Stebėsenos posistemė sujungta su SPS posisteme, kuri gali aptikti stebimo objekto aprašomų duomenų nuokrypius. Kai nuokrypiai viršija aprašomas ribas, informuojama perduodama stebėsenos posistemei. Laikas gauti tinkamą sprendimą dažniausiai būna ribotas. Sistemos veikimas apibrėžiamas laikina priklausomybe, dinamišku situacijos vertinimu ir prisitaikant prie grįžtamojo ryšio kontrolės, kuri turi būti įgyvendinta įterptinėje SPS posistemėje.

Stebėjimo sistemos komponentas integruoja keletą sensorių sistemų, kurios stebi ir rodo galimus parametrus. Tokių sensorių tikslas – buvimo vietoje stebėti objekto fizinius parametrus. Kokiu būdu sensorio tipai galimi integruoti SPS yra pavaizduota 4.3 pav. Keletas svarbių sensorių pavyzdžių: įmontuota kamera, judesio sensoriai, vietovės pozicijos duomenis (GPS įranga), naudojami objektui stebėti.



4.2 pav. SPS pagrindinių komponentų architektūra

Sensorių duomenys pristatomi SPS sistemai, kuri diagnozuoja susidariusią būseną. Sesijos inicijavimo protokolą taikysime kaip pagrindinį protokolą duomenų apsikeitimui. Sensorių posistemė dirba kaip agentas ir lygiagrečiai surenka duomenis. Procesų valdymo posistemė SPS turi aptikti tokius faktus, kaip nustatytų leistinių reikšmių konkrečiu laiko momentu viršijančius duomenis (leistiną medžiagos temperatūrą, smūgio jėgą ir kt.) (Xie, 2011) ir pagal tai siųstų įspėjimo žinutes reikiamoms tarnyboms.

Sensoriai gali būti skirstomi į tam tikras grupes. Dažniausiai naudojami sensorių tipai: fiziniai ir virtualūs sensoriai. Aparatūrinės įrangos sensoriais galima užfiksuoti beveik visus fizinius duomenis. Yra galimybė integruoti tokius sensorių jutiklius į sistemą:

- šviesos – fotodiodų, spalvų sensoriai;
- vizualizacinis kontekstas – įvairios kameros;
- garsinis – mikrofona;
- judesio – akcelerometrai, judesio detektoriai, magnetiniai laukai;

- vietos – GPS;
- temperatūros – termometrai;
- galios – galios sensoriai;
- virtualūs sensoriai – šaltinio kontekstiniai duomenys gaunami iš skirtingų paslaugų ar programinės įrangos. Virtualūs sensoriai gali būti elektroninis kalendorius, naršyklė ir kt.

4.3. Kelio ruožų rizikos ypatumų aprašymas SPS sistemoje

Norint aprašyti transportavimo kelią, maršrutą suskaidome į kelio ruožus ir kiekvieną jų aprašome skirtingomis charakteristikomis. Pagal (Fabiano, 2005, Brauers, 2008), galima aprašyti tikėtiną eismo įvykių skaičių kelio ruožo r atkarpoje, kuris vystysis pagal scenarijų s_k ir jį galima išreikšti taip:

$$B_r = \sum_{s_k} f_r N_{r,s_k} P_{s_k} \quad (1)$$

kur f_r yra ne laimingų įvykių dažnis r -tame kelio ruože [incidentai \cdot metai⁻¹], N_{r,s_k} yra tikėtinas eismo įvykis pagal scenarijų s_k , r -ame kelio ruože [incidentai⁻¹], P_{s_k} yra scenarijaus S_k tipo tikimybė, pagal priskirtą nelaimingą įvykį (susidūrimas, apvirtimas, ar gedimas). Nagrinėjama galimų scenarijų aibė $S = \{s_k\}$.

Tikėtinas nelaimingų įvykių skaičius, visame kelio ruože:

$$B = \sum_r \sum_k f_r N_{r,s_k} P_{s_k} \quad (2)$$

Nelaimingų įvykių dažnis pagal s_k scenarijų r -ame kelio ruože apskaičiuojamas:

$$f_{r,s_k} = f_r \cdot P_{s_k}, \quad f_r = \gamma_r L_r n_r, \quad (3)$$

kur $\gamma_r = \gamma_{0,r} G$ yra tikėtinas dažnis r -ame kelio ruože [nelaimingas įvykis \cdot km⁻¹ \cdot transporto priemonės⁻¹ \cdot metai⁻¹], L_r yra kelio ilgis [km], n_r yra transporto priemonių kiekis per kelio atkarpą r -ame kelio ruože [transporto priemonės], $\gamma_{0,r}$ yra nelaimingų įvykių dažnis [incidentas \cdot km⁻¹ \cdot transporto priemonės metai⁻¹].

G yra tikimybinis aplinkos vertinimo parametras. Įvairių veiksnių daroma įtaką nelaimingiems įvykiams, kurie gali būti šie: mechaniniai, aplinkos, elgesio, fiziniai.

Bendrai šie parametrai aprašomi taip:

$$G = \prod_{j=1}^m G_j. \quad (5)$$

G yra stiprinantis ar lengvinantis parametras darantis įtaką nelaimingiems įvykiams. Pagrindinius šių parametrų tipus galima apibūdinti taip: G_1 yra parametras, priklausantis nuo temperatūros; G_2 yra parametras, priklausantis nuo kelio ruožo būdingų faktorių, tokių kaip: tunelis, kelio posūkis, įkalnė ir kiti parametrai; G_3 yra parametras, priklausantis nuo metrologinių veiksnių, tokių kaip sniegas, lietus, ledas ir kiti; G_4 yra parametras, priklausantis nuo vėjo greičio ir vėjo krypties ar kitų parametrų, kuriuos mes priskiriame kaip G_m . N_{r,S_k} yra bendras nelaimingų įvykių skaičius pagal (2):

$$N_{r,S_k} = (\Phi_{S_k}^{in} o^{\Delta t} v_r + \Phi_{S_k}^{off} d_r) P(F, S_k) \quad (6)$$

Buvimas kelyje ir už kelio ribų nelaimingų atsitikimų skaičius apskaičiuojamas atitinkamai:

$$N_{r,S_k}^{inn} = \Phi_{S_k}^{in} o^{\Delta t} v_r P(F, S_k) \quad (7)$$

$$N_{r,S_k}^{off} = \Phi_{S_k}^{off} d_r P(F, S_k), \quad (8)$$

kur $\Phi_{S_k}^{in}$ yra scenarijaus S_k pasekmė esant kelio ruože [m^2];

$\Phi_{S_k}^{off}$ yra scenarijaus S_k pasekmė už kelio ruožo ribų [km^2];

$P(F, S_k)$ yra tikimybė, kad įvykis nelaimingas įvykis F , pagal scenarijų S_k ;

$o^{\Delta t}$ yra vidutinis transporto priemonių užimtumo veiksnys per tam tikrą laikotarpį;

d_r gyventojų tankumas r -ojo kelio ploto aplinkoje [$gyventojai \cdot km^{-2}$];

v transporto priemonių tankis kelio plote [$transporto priemonė \cdot m^{-2}$];

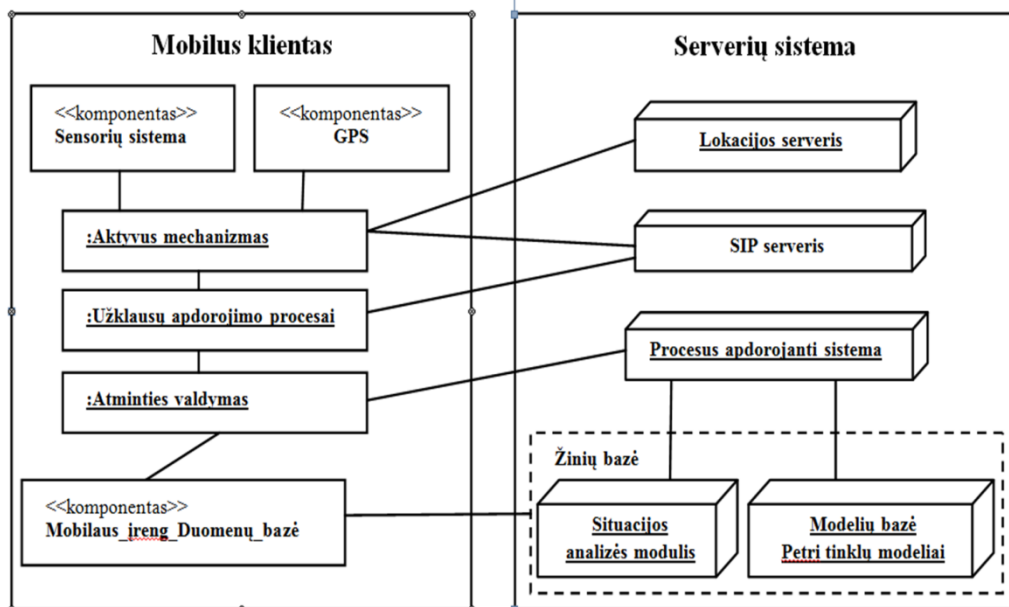
k yra scenarijaus skaičius.

4.4. Transporto objektų buvimo vietos duomenų atvaizdavimas, pasinaudojus mobiliosiomis technologijomis

Judantys objektai (mobilūs klientai) gali būti apriboti tinklo infrastruktūra, gaunant informacijos apie jo poziciją ir kartais tik iš GPS imtuvų pagal vietos informaciją galima gauti esamą poziciją. Lokalizacijos serveris ir centralizuota duomenų saugykla yra serverio pusėje, o duomenų apsikeitimas yra įmanomas belaidžiais tinklais (Booth, 2004). Išplečiama serverio dalies architektūra, kuri leidžia gauti duomenis mobiliuoju telefonu per interneto paslaugas. Pagrindinį vaidmenį atlieka sesijos inicijavimo protokolas – SIP ir procesus apdorojanti komponentė, kuri valdoma žinių bazės modelių pagalba. Žinių bazėje saugomi Petri tinklų modeliai leidžia įjungti procesus valdančias taisykles į bendrą sistemą šių sudėtingų procesų atpažinimui ir valdymui. Šios sistemos architektūros infrastruktūroje bandyta spręsti duomenų integravimo tarp heterogeninių sistemų ir jų dalinio suderinimo problemas.

Sujungimas tarp kliento ir serverio yra realizuojamas tinkle, kai objektas pradeda judėti. Sujungimo įvykis aktyvuoja mechanizmą su pranešimu serveriui, nurodančiu, kuriuos tinkamus veiksmus reikia atlikti. Duomenys yra išsaugomi duomenų saugykloje po kiekvieno judančio objekto pozicijos atnaujinimo. Judančio objekto pozicija atnaujinama duomenų saugykloje tada, kai objekto reali pozicija gaunama iš GPS imtuvo ir nesutampa su pozicija, esančią duomenų saugykloje.

Klientas gauna savo vietovės koordinates iš GPS imtuvo ir iš fizinių ar virtualių sensorių, tai leidžia rinkti duomenis iš sensorių ir internetu juos apdoroti. Jei sensoriaus duomenys pasikeičia už nustatytų ribų, SPS gauna signalą ar žinutę apie pasikeitimus. Šių komponentų architektūra pavaizduota (žr. 4.4 pav.).



4.4 pav. Bendroji įrenginio ir serverio sąveikos komponentų architektūra

Laikina duomenų bazė saugo duomenų įrašus fiksuotu laiko momentu ir praėjus laiko tarpui išsiunčia į pagrindinę duomenų saugyklą. Laikina duomenų bazė dažniausiai naudojama realaus laiko stebėjimui ir kai prarandamas ryšys su pagrindine duomenų saugykla.

Antrasis sluoksnis yra atsakingas už kontekstinių duomenų surinkimą. Šis sluoksnis naudoja fizikinių sensorių tvarkykles ir virtualių ar loginių sensorių programinę įrangą. Užklausų funkcionalumas įgyvendina daugkartinio panaudojimo programinės įrangos komponentus, tokius kaip getPosition(). Vietos nustatymo paslaugas apribotiems mobiliesiems įrenginiams galima nustatyti pasinaudojant J2ME programavimo sąsaja. Vietos nustatymo paslaugą galima gauti praplėčiant:

```
Criteria crit = new Criteria();
crit.setHorizontalAccuracy(500);
LocationProvider locpro = LocationProvider.getInstance(crit);
Location loc = locpro.getLocation(60);
Coordinates cor = loc.getQualifiedCoordinates();
if (cor != null) (
```

```
double lat = cor.getLatitude();  
double lon = cor.getLongitude();
```

Kur klasė (Criteria) naudojama tikslumui nustatyti. Pavyzdžiui Criteria crit = new Criteria (); cr.setHorizontalAccuracy(500) reiškia, kad horizontalus tikslumas yra 500 metrų.

Ketvirtas sluoksnis (saugykla ir valdymas) organizuoja surinktus duomenis ir suteikia klientams priėjimą prie tų duomenų per viešą sąsają. Klientai gali gauti duomenis dviems skirtingais būdais: sinchroniniu ir asinchoriniu.

Naudojantis mobiliosiomis žiniatinklio paslaugomis P2P pagrindu, komunikacijos tarp įrenginių užtikrinimui naudojamas sesijos inicijavimo protokolas. Mobiliosios žiniatinklio paslaugos galutinis adresas yra SIP URI (Xie, 2011).

Suderinti žiniatinklio protokolą SOAP su SIP svarbu dėl komunikacijos saugumo tarp serverio ir mobilių įrenginių. SOAP transporto neutralus mechanizmas naudojamas kartu su SIP tame pačiame sluoksnyje (Imran, 2011).

Petri tinklo schema išreiškianti judančio objekto parametrų sekimo procesą pateikiama 4.3 pav. Kad fiksuoti gaunamų parametrų sekas belaidžio tinklo protokole, įvedami papildomi parametrai, numatomi SPN aprašyme: colset - spalvų rinkinys, NO = *int* - spalvą atitinkančio parametro tipo numeris. NO naudojamas modeliuoti veiksmų sekos eiliškumą protokole.

Spalvų rinkinio ir duomenų Dekarto sandauga NOxDATA išreiškia gaunamų duomenų ir jų prieskyrų skirtingiems duomenų tipams sąveiką. Čia pirmas narys yra pozicijos eilės numeris, antrasis narys fiksuojami duomenys išreiškiami teksto eilute.

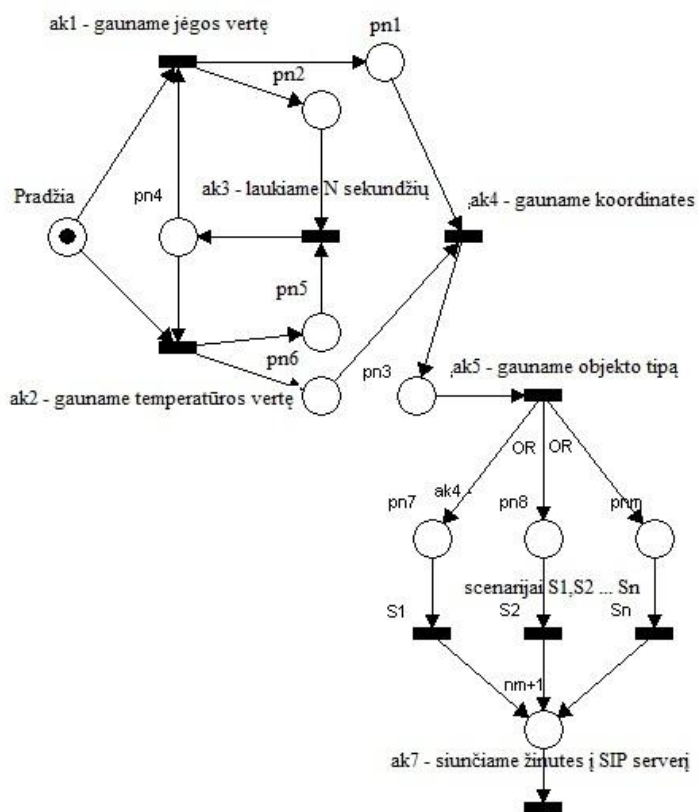
Spalvų rinkiniai aprašomi pagrindiniais parametrais taip:

```
colset DATA = string;  
colset NOxDATA = product NO * DATA.
```

Spalvos rinkinys DATA yra naudojamas, kad modeliuoti naudingąjį duomenų paketų apkrovą. Spalvų rinkinys NOxDATA naudojamas modeliuoti duomenų paketus, kur taikomi duomenys yra eilės numeris ir patys duomenys.

CPN priemonėmis perduodamų duomenų paketų aprašymas belaidžiuose tinkluose pateikiamas 4.3 pav. Kiekviena žymė turi savo semantinę prasmę ir vertę modelyje. Inicijuojant sesiją kiekvienos žymės aprašas pateikiamas taip:

- reiksme1: INVITE sip:B_Vartotojas@serveris.org SIP/2.0;
- reiksme2: Via: SIP/2.0/UDP lab.serveris.org:5060;
- reiksme3: To: B_Vartotojassip:B_Vartotojas@radio.org;
- reiksme4: From: A_Vartotojassip:A_Vartotojas@serveris.org;
- reiksme5: Call-ID: 123456789@lab.serveris.org;
- reiksme6: CSeq: 1 INVITE;
- reiksme7: Subject: Testinis pranešimas...
- reiksme8: Contact: sip:A_Vartotojas@serveris.org;
- reiksme9: Content-Type: application/sdp; Content-Length: 150



4.3 pav. Petri tinklo schema išreiškianti judančio objekto parametrų sekimo procesą

4.5. Ketvirtojo skyriaus išvados

Šiame skyriuje pasiūlyta sprendimo paramos sistemos architektūra, kuri integruoja sesijos inicijavimo protokolo išplėtotas funkcines galimybes ir įgalino įterptinių sistemų pagalba vykdyti sensorių monitoringo duomenų saugojimą nutolusiose duomenų saugyklose ir pritaikyti vieningą duomenų magistralę, kaip integruotą SIP funkciją duomenų apsikeitimui vykdyti transporto judančių nutolusių objektų monitoringą (stebėseną) ir vertinimą.

5. SIP INTEGRACINIŲ GALIMYBIŲ VERTINIMAS EKSPERIMENTIŠKAI APJUNGIANT SKIRTINGAS HETEROGENINES SISTEMAS

Šiame skyriuje aprašomas eksperimentinis tyrimas, padedantis įvertinti Sesijos inicijavimo protokolo integracijos galimybes, apjungiant į bendrą magistralę keturių tipų įrenginius. Eksperimentiniam tyrimui atlikti sukonstruota prototipinė sistema, kuri leis užtikrinti duomenų perdavimą tarp keturių tipų įrenginių ir įvertinti vieningo protokolo t.y. SIP galimybes. Skyriuje pateikti eksperimentinių tyrimų rezultatai, kurių tikslas yra eksperimentiškai patvirtinti magistralės panaudojimą duomenų apsikeitimui tarp nutolusių įrenginių, diagnozuojant pavojingų objektų būsenas ir suteikiant atgalinį ryšį jų valdymui. Skyrius sudarytas iš šių dalių: eksperimentinių tyrimų sąlygų ir duomenų aprašymo, pasiūlytos techninės ir programinės įrangos komplekto architektūrinių sprendinių aprašymo. Pabaigoje aprašoma, kaip nutolusius objektus sujungti į bendrą sistemą, naudojant vieningą komunikavimo protokolą, ir pateikiami eksperimentinių tyrimų rezultatai.

5.1. SIP išplėtimo eksperimentinio tyrimo sąlygų aprašas

Technologinė įranga, skaitanti duomenis iš sensorių, yra pagrindinis komponentas, perduodantis duomenis į duomenų saugyklas. Sensorinės įrangos sensorių atitinkamų parametrų reikšmės gali būti perduodamos į nutolusius serverius, kuriuose šie duomenys yra sisteminami ir pritaikomi analizei.

Technologinė įranga, skaitanti duomenis iš sensorių, yra paskirstytos sistemos dalis, kuri geba nuskaityti parametrinius duomenis iš nutolusių heterogeninių įrenginių. Nutolę objektai palaiko skirtingus protokolus, kuriuos sujungsime į vieningą protokolą.

Sesijos inicijavimo protokolas gali būti naudojamas kaip signalizavimo protokolas, galintis apimti įvairias paslaugas skirtingoje aplinkoje. SIP leidžia greitai apdoroti pranešimus, kas yra labai svarbu įvairiems įrenginiams, pavyzdžiui SIP telefonams ir įvairiems tiltams (KUMAR Singh, 2008, Yang, 2009). Siūlomas bendras ryšys tarp išskirstytų įrenginių. Skirtingus įrenginius

kontroliuoti naudojama SIP protokolo komunikacijos magistralė, įskaitant Zigbee, RS232, X10 ir kitus ryšio įrenginius.

5.2. Eksperimentinių tyrimų uždaviniai

Nutolusių įrenginių (sensorių) komunikavimo protokolus pakeisti ir sujungti vieningu protokolu. Vartotojui nereiks kurti daugybės skirtingų dalykinių programų nuskaitant sensorinius duomenis.

Pasiūlyti sprendimo palaikymo komponentų kūrimo metodiką.

Išnagrinėti intelektualių komponentų galimybes, kurios būtų integruojamos į paslaugas arba jungtis tarpusavyje ir pasiūlyti panaudoti sprendimų paramos sistemą architektūroje.

Sukurti prototipą, taikant imitacinį modeliavimą ir žinių valdymo komponentus, juos sujungti bei išbandant.

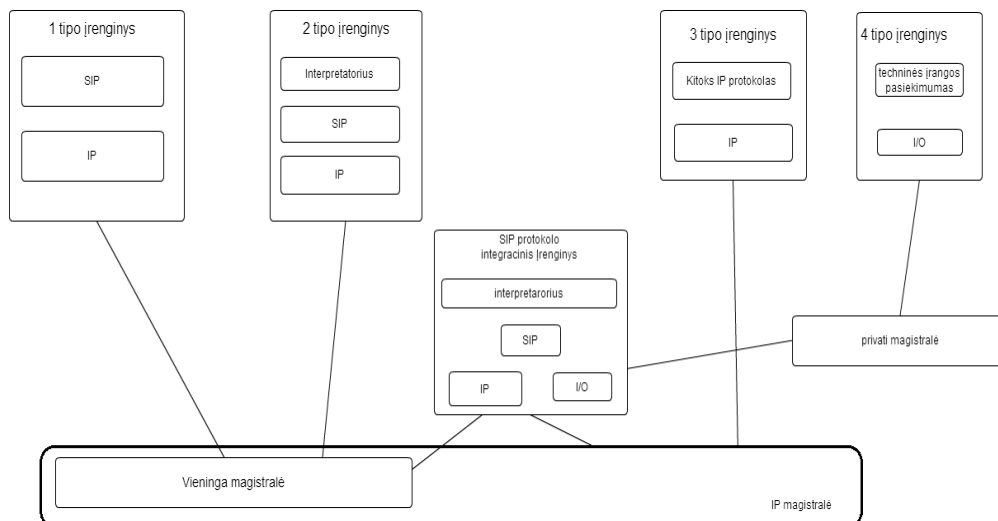
5.3. Eksperimentinė platforma

Pasiūlyta universali komunikacijos magistralė, kuri įgalina stebėti objekto vietą ir skirtingas būsenas skirtingais laiko momentais. Eksperimentinė platforma susideda iš atmega plokštės 16 MHz skaičiavimo dažniu ir 4Kb vidinės atminties ir sujungta su tinklo moduliu siusti SIP žinutes interneto tinklu. SIP žinučių valdymui pasirinkta aplikacijų serveris Tomcat ⁷ ir integruotos bibliotekos SIP žinučių valdymui. Media valdymui į aplikacijų serverį Tomcat 7 integruotas media serveris „Mobicents Media Server 2.1“. Pagrindinis tikslas yra pasiūlyti platformą, kuri būtų integruota į paplitusią VOIP platformą, t.y visi įrenginiai bendrautų SIP pagrindu. Šioje aplinkoje yra įdiegti įvairūs įrenginiai, nuo telefonijos įrangos iki smulkiausių sensorių, matuojantys aplinkos parametrus. Šiuos įvairius įrenginius ir sensorius apjungti reikalingas interpretatorius. Interpretatoriui projektuoti pasirinkta Atmega plokštė ir buvo pasinaudota oSIP dalykinės programos programavimo sąsaja (Application Programming Interface API). Tai sąsaja, kurią suteikia kompiuterinė sistema, biblioteka ar programa tam, kad programuotojas per kitą programą galėtų pasinaudoti jos funkcionalumu ar

² <http://www.mobicents.org/installation-tomcat.html>

apsikeistų su ja duomenimis. Šia programavimo sąsaja pasinaudojus buvo rašoma programinė įranga. Jos pagalba sensorių parametrų duomenys buvo konvertuojami į sesijos inicijavimo protokolo standartus atitinkančias žinutes interpretatoriuje. Programinė įranga interpretatoriuje parašyta C kalba. oSIP palaiko pagrindinius transportavimo protokolus, tokius kaip TCP, UDP ir TLS. oSIP programinė įranga yra parašyta C kalba ir nereikalauja kitų bibliotekų priklausomybės, išskyrus standartinę C biblioteką. Programinei įrangai realizuoti buvo pasinaudota libosip2parser³ biblioteka, kuri buvo modifikuota, kad suteiktų galimybę sukompiliuoti ir turėtų interpretatoriaus funkcionalumą Atmega plokštėje. Suteikiama galimybė įrenginiams, palaikantiems SIP funkcionalumą, bet neturintiems tiesioginio sujungimo su išskirstytomis sistemomis, pasinaudoti SIP tarpinio serverio galimybėmis.

Aukščiau neginėtą platformą galima naudoti kaip priemonę eksperimentuoti su įvairiais scenarijais. Šiam tikslui sukurtas sistemos prototipas, kuris apima išmaniuosius telefonus, IP kameras, ZigBee⁴ sensorius ir X10 kontrolierius. Keturių skirtingų įrenginių tipų apjungimas į vieningą protokolą (5.1 pav.).



5.1 pav. Skirtingų įrenginių tipų apjungimo platforma

Naudojamas sesijos inicijavimo protokolas komunikacijos magistralėje duomenų apsikeitimui tarp sensorių ir išskirstytų įrenginių. Darbe išbandytas sensorių prisijungimas prie SIP ryšio magistralės. SIP adapteriui turime suteikti

³ <http://www.gnu.org/s/osip/>

⁴ <http://www.zigbee.org/>

galimybę pasinaudoti tinkamais funkcionalumais per SIP priėmimo mechanizmus. Kad pasiektume šiuos funkcionalumus yra apibrėžti trys sąveikos režimai, pasiekiami Sesijos inicijavimo protokolu:

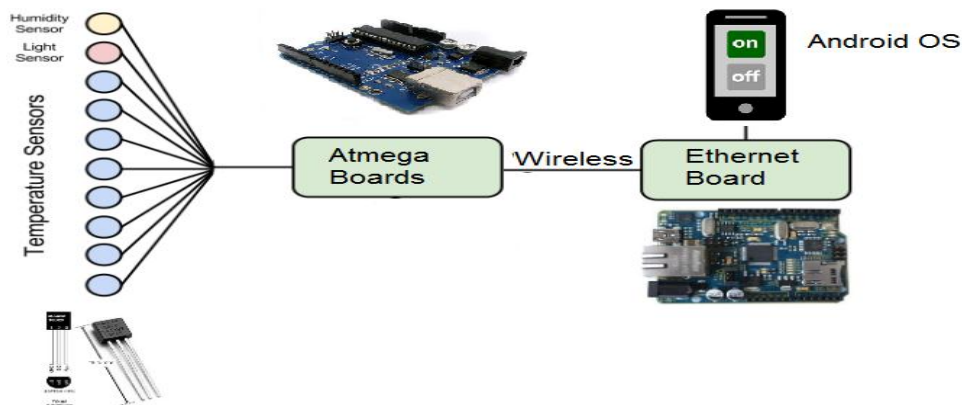
- komandos režimas – kontroliuoti įrenginius ir statusų užklausas;
- įvykių valdymo režimas (įvykių skelbimas ir įvykių skelbimų užsakymas);
- sesijos režimas (kvietimas į duomenų srauto sesijas).

Užklauso ir gavimo duomenys turi palaikyti skirtingus formatus: komandų parametrų reikšmes (pvz.: SOAP), įvykių vertes (pvz.: xml formatas) ir sesijos galimybių aprašą (naudojantis SDP tekstu).

SIP adapteris teikia interpretatoriaus funkcionalumą.

SIP komunikacijos įgalinimas susideda iš dviejų pagrindinių dalių:

1. Aparatūrinė platforma susideda iš sensorių ir keitiklių, kurie sujungti į įterptinės sistemos interneto pralaidos plokštę su tinklo galimybėmis;
2. Specifinė programinė įranga įterptinėje sistemoje, galinti bendrauti sesijos inicijavimo protokolu ir bendrauti su skirtingais sensoriais (žr. 5.2 pav.).



5.2 pav. SIP komunikacijos magistralės iliustracinė schema

Sensorių įgalinimas bendrauti su sistemomis tilto (gateway) pagalba, kuris susideda iš:

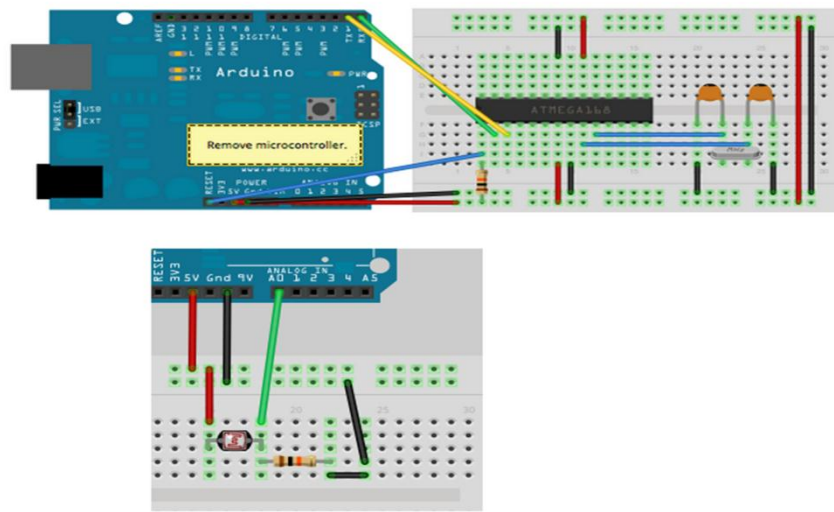
Atmega mikrokontrolerio, kuris yra fiziškai sujungtas (USB sąsaja ar nuoseklia jungtimi) ir internetinė plokštė.

Sensoriai sujungti nuoseklia jungtimi. Mikrokontroleryje Atmega suprogramuota programinė įranga, kuri leidžia surinkti duomenis iš sensorių ir perduoti sistemoms SIP specifikaciją atitinkančiomis žinutėmis.

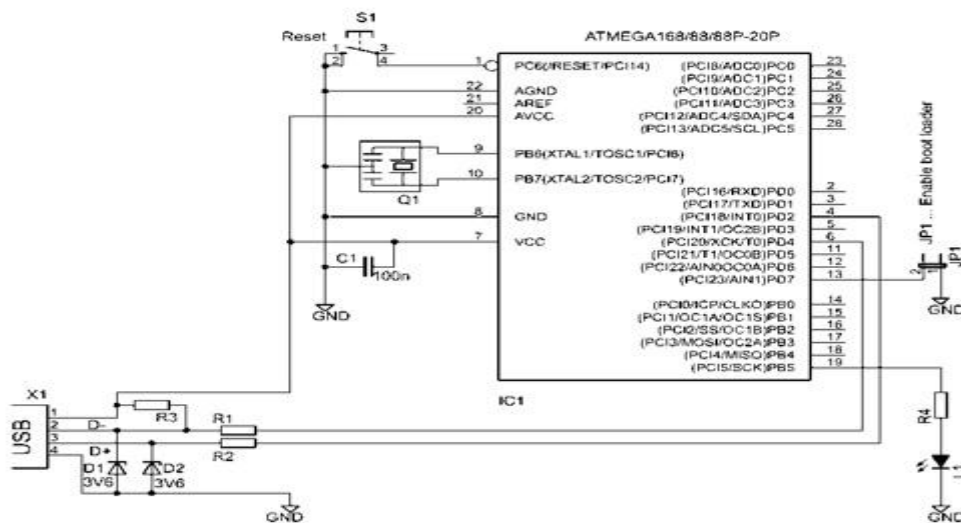
Valdymo pultas, sudarytas iš šių komponentų: jungtys, varžos ir kt. (5.3 pav.)

Atmega kontrolerio jungčių struktūra pateikta 5.4 pav.

SIP komunikacijos įgalinimui tarp skirtingų nutolusių sensorių, reikia dviejų pagrindinių komponentų: aparatūrinės įrangos platformos, kuri tiesiogiai prijungta prie sensorių, ir pačių sensorių. Aparatūrinės įrangos mikroschemos Atmega jungčių ir komponentų schemas pavyzdys pateiktas 5.3 pav.



5.3 pav. Aparatūrinės įrangos mikroschemos skirtos SIP valdymo pultui maketo pavyzdys



5.4 pav. Mikrovaldiklio plokštės schema

Mikrovaldiklis (sutr. MK) yra priskiriama vienakristalių mikro skaičiavimo mašinų klasei, kurios gali būti: Atmega, PIC ar kitos panašios mikroschemos (žr. 5.4 pav.).



5.5 pav. Mikroschemos Atmega jungčių schema

Valdiklis - plokštė, kuri yra specializuotos paskirties, nesavarankiškas kompiuteris, kurioje įlituota MK mikroschema. Valdiklis yra elektroninis įtaisas, kontroliuojantis ir reaguojantis į pasikeitusius aplinkos parametrus. Aplinkos parametrai nuskaitomi specialiais davikliais. Valdymo įtaisai (elektroniniai moduliai bei komponentai, motorika) – tai, ką galima valdyti paduodant elektros impulsus.

Maitinimo jungtis (pins) yra keičiama taip: VIN – įėjimo įtampa į plokštę, kada naudojame išorinį maitinimo šaltinį (pvz., 5 voltų maitinimas iš USB jungties ar kiti šaltiniai, t. y. baterijos). Jei įtampos maitinimas yra elektros lizdas, tada naudosime šią VIN jungtį. 5V yra reguliuojamas maitinimo šaltinis, kuris

naudojamas aprūpinti elektros energija mikrokontrolerį ir kitus plokštės komponentus. Taip pat energiją galima teikti per VIN jungtį, per plokštės reguliatorius, per USB jungtį ar per kitą reguliuojamus 5V teikiančią energiją jungtį. Sukurtos plokštės reguliatorius pateikia 3V energiją, kurios maksimali leistina srovė yra 50mA. Taip pat mikrokontrolerį sudaro GND – įžeminimo jungtis, skaitmeninės įėjimo/išėjimo jungtys. Iš viso yra 14 jungčių (kurių 4 yra PWN išėjimai (PWN yra analoginė išėjimo jungtis)).

Atmega – tai vienas iš valdiklių, kuriuo galima priversti įrenginį atlikti numatytus darbus. Prie Arduino valdiklio plokštės galima jungti daugybę periferinių įrenginių – daviklius (jėgos, temperatūros, pagreičio ir t. t.), skystų kristalų (LCD) ekranėlius, servo variklius. Šiuos valdiklius užprogramuojant, galima priversti reaguoti į pasikeitimus ar vykdyti atitinkamus veiksmus. Pridėjus GSM/GPRS modulį galima siųsti duomenis internetu ar SMS žinutes tiesiogiai į numatytą telefoną. Pridėjus GPS modulį, suteikiama galimybė nuskaityti esamos vietos koordinatas. *Atmega* valdiklis pasirinktas dėl šių priežasčių:

- Tai yra atviras projektas, kurį galima nesunkiai išbandyti eksperimentiškai;
- paprastas *Atmega* komunikavimas su kompiuteriu;
- daugumai modelių nereikalingas papildomas programatorius;
- gali būti taikomos atviros programavimo bibliotekos, schemas.

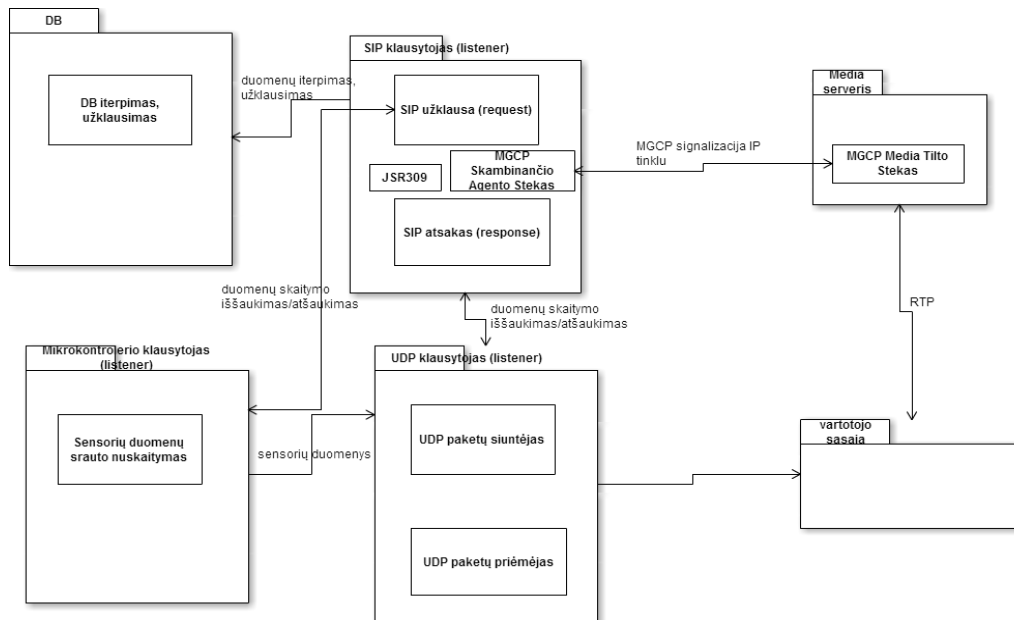
Naudojami sensoriai yra šie: temperatūros sensorius ir analoginis atstumo nustatymo sensorius. Aplinkos oro temperatūrai nuskaityti naudojamas puslaidininkinis LM35 temperatūros sensorius (žr. 5.6 pav.). Sensoriaus matavimų ribos yra šios:

- Tipas: Analoginis
- Jautrumas: 10mV laipsniui (matuojamas Celsijais)
- Funkcionavimo ribos: -40 laipsniai pagal Celsijų iki 150 laipsnių Celsijų

pagalba išmaniuoju telefonu galima pasiekti sensorių duomenis, kaip pateikta 5.8 pav. Duomenų siuntimas ir gavimas vyksta SIP protokolo pagrindu (žr. 5.8 pav.). Šioje kuriamoje sistemoje buvo išbandytas įvairių prijungtų objektų veikimas. Buvo pajungtas išmanus telefonas, judesio sensoriai, temperatūros sensoriai ir kt. Ši platforma galėtų būti pradėta naudoti transporto priemonėse, išbandant įvairius scenarijus ir tiriamų pavojingų krovinių transportavimui. Pagrindiniai platformos komponentai:

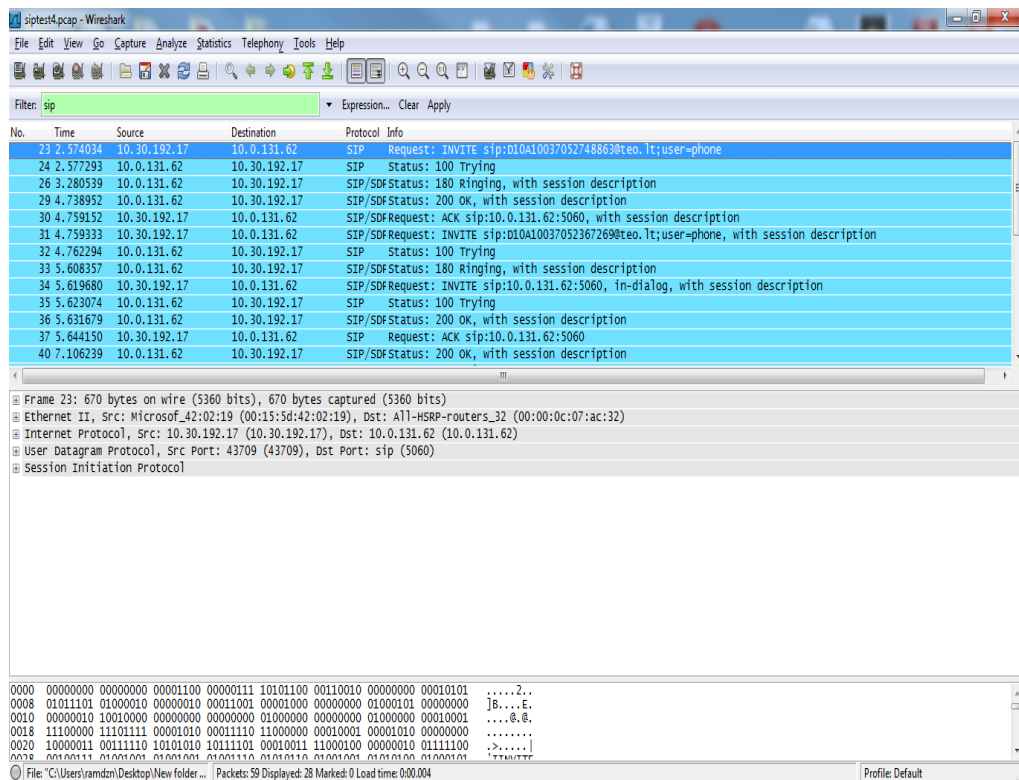
- *Media serveris* yra specializuota programinė įranga skirta saugoti įvairius skaitmeninius laikmenis (skaitmeninę video, garso / muzikos ir vaizdo informaciją ir failus).
- *SIP klausytojas* – komponentas, skirtas sukurti komunikaciją tarp vartotojo ir media serverio teikiant galimas paslaugas, t.y. skaitmeninių laikmenų išskvietimas ir saugojimas.

Šių komponentų veikimas aprašomas programavimo kalba integruojant naudojamus įrenginius į SIP komunikaciją, suteikiant vieningos duomenų mainų magistralės komunikacijos galimybes tarp išskirstytų įrenginių.



5.8 pav. Siūlomos sistemos komponentų sąveikos struktūra užtikrinant SIP komunikaciją

Duomenų srauto sekimui ir paketų analizei naudojamas „Wireshark“ paketų ir protokolų analizatorius. Naudojant šią programą galime įrašyti ir žiūrėti paketus keliaujančius tinkle, matyti jų turinį ir nustatyti protokolo tipą. Įrankis, leidžiantis stebėti žinučių siuntimo būklę realiaame laike ir atvaizduojantis jų siuntimo seką ir SIP protokolo veikimo parametrus, yra pavaizduotas 5.9 pav.



5.9 pav. Realiaame laike stebimų žinučių siuntimo būklės stebėjimo ekranas

Paketų analizatorius naudojamas šioms tikslams: analizuoti saugumo ir problemas tinkle; tyrinėti, kurti, tobulinti protokolus, taip pat naudojama mokymosi tikslams.

5.4. Eksperimentinių tyrimų rezultatai

Pasiūlyta intelektualią paslaugų teikimo dinaminėje aplinkoje sistema, kurioje taikomi žinių valdymo metodai, mobiliųjų išskirstytųjų sistemų sąveikos metodai ir mobiliosios technologijos.

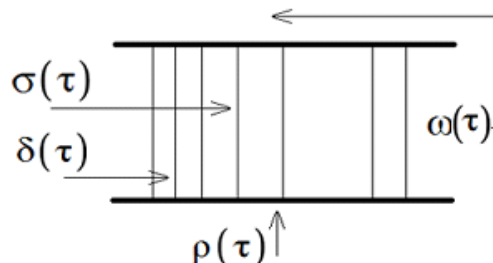
Pateiktos priemonės (metodai ir jų taikymo metodika) leidžia efektyviau pasinaudoti žinių valdymo baze, įjungiant intelektualius komponentus į bendrą

sprendimų paramos sistemos architektūrą, kurios rezultatas – paslaugų tiekimas nuotolinio bendradarbiavimo aplinkoje, valdant dinamiškai kintančią aplinką.

Sistema palaiko ryšį tarp judančių objektų ir leidžia perduoti heterogeninius duomenis, kur komunikacijai užtikrinti klasifikuojami objektai. Išnagrinėta, kaip objektai turi būti pritaikyti, kad prisijungtų prie SIP ryšio magistralės.

5.5. Eksperimentinės sistemos SIP žinutes apdorojančių įrenginių apkrovos tyrimas

SIP tinklas susideda iš daugybės geografiškai išdėstytų nukreipiančių serverių ir daugybės vartotojo agentų. Kiekvienas nukreipiantysis serveris yra atsakingas už ryšio nustatymą tarp dviejų vartotojo agentų. Šis serveris persiunčia užklausas ir sukuria laikinus atsakus, patvirtinančius kiekvieną gautą užklausimą iš siuntėjo vartotojo agentų – tai gali būti inicijuojantis UA ar nukreipiantysis serveris. Retransliacijos mechanizmas reikalingas, užtikrinti patikimus SIP žinučių pristatymus. Tačiau vienu metu gaunant per daug inicijuojamų pirminių SIP INVITE žinučių susidaro nereikalingas apdorojimo uždelsimas. Neapdorojant pradines SIP INVITE žinutes, serveriui siunčiamos retransliuotos žinutės, dėl kurių susidarymo paspartėja perkrova, tai galiausiai apkrauna visą tinklą (Luo, 2008; Li, 2010), todėl būtina aprašyti tinklo apkrovos dinamiką perkrautiems SIP serveriams (Rafique, 2009). Bet kuriuo metu susidarius perkrovai tinkle, vienas serveris būna labiau apkrautas nei kiti serveriai, todėl norint maksimaliai sumažinti žinučių praradimus SIP tinkle, reikia visus inicijuojančius serverius nukreipti per nukreipiantįjį serverį.



5.11 pav. SIP žinučių eilių dinamikos iliustracinis pavyzdys perkrautuose SIP serveriuose.

Tinklo apkrovai aprašyti taikysime matematinę eilių teorijos modelį, kuris dažnai taikomas panašaus tipo uždaviniams spręsti. Grafiškai tinklo apkrovų galimybės pavaizduotos 5.11 pav. Šiame paveikslėlyje $\rho(\tau)$ reiškia bendrą tinklo apkrovą laiko momentu τ . Rodyklės parodo, kad laiko momentu τ į serverius ateina $\sigma(\tau)$ naujų (INVITE tipo) žinučių, bei $\delta(\tau)$ retransliuotos žinutės laiko momentu τ , kurios atsiranda dėl to, kad serveris negalėdamas aptarnauti ateinančių užklausų, jas atideda vėlesniam laikui. Tuo tarpu $\omega(\tau)$ rodo, kiek laiko momentu τ serveris pajėgia apdoroti žinučių. Visą tai apibendrinus galima užrašyti eilių teorijoje neretai sutinkamą lygtį, kuri aprašo, koks yra tinklo apkrovimas laiko momentu $\tau + 1$

$$\rho(\tau + 1) = [\rho(\tau) + \sigma(\tau) + \delta(\tau) - \omega(\tau)]_+, \quad \tau = 0, 1, 2, \dots \quad (1)$$

kur „+“ parodo tai, kad tinklo apkrova negali būti neigiamas dydis.

Pagal SIP specifikaciją, iš viso gali būti ne daugiau nei 6 pradžioje iniciuotos retransliuotos žinutės. Todėl galima apskaičiuoti suminį retransliuotų žinučių skaičių $\delta(\tau)$ laiko momentu n pagal formulę.

$$\delta(n) = \sum_{j=1}^6 \delta_j(n),$$

čia $\delta_j(n)$ rodo, kiek originalių žinučių atėjusių laiko momentu $n - T_j$ buvo retransliuota laiko momentu n , $T_j = (2^j - 1)T_1$ ir $1 \leq j \leq 6$.

Laiko momentu $n - T_j$ naujai atėjusių žinučių skaičius buvo lygus $\sigma(n - T_j)$, o tinklo apkrova buvo lygi $\rho(n - T_j)$. Kadangi perkrautas serveris gali apdoroti $\sum_{k=1}^{T_j} \omega_j(n - T_j + k)$ žinučių per T_j laiko tarpą, likusių neapdorotų žinučių skaičius tampa lygus

$$\left[\sigma(n - T_j) + \rho(n - T_j) - \sum_{k=1}^{T_j} \omega_j(n - T_j + k) \right]_+,$$

kuris negali būti neigiamas. Kadangi laiko momentu n reikia retransliuoti tik originalias laiku $n - T_j$ atėjusias žinutes, tai naudojama min funkcija

$$\delta_j(n) = \min \left\{ \left[\sigma(n - T_j) + \rho(n - T_j) - \sum_{k=1}^{T_j} \omega_j(n - T_j + k) \right]_+, \sigma(n - T_j) \right\}.$$

Pateiktos lygtys rodo perkrauto SIP serverio dinaminę elgesį (daugiau žr. Hong 2010).

Žinučių retransliacija gali suteikti patikimą SIP žinučių perdavimą, bet tuo pačiu jis padidina serverio apkrovą. Dėl šios priežasties atsiranda serverio perkrova. Hong 2010 išnagrinėjo, kokias teorines sąlygas turi atitikti pradiniai duomenys, kad apkrauta sistema išliktų stabili. Nemažinant bendrumo tariama, kad žinutės ateina pastoviu režimu $\sigma(\tau) = \sigma$, žinutės taip pat apdorojamos pastoviu režimu $\omega(\tau) = \omega$, naujai gautos žinutės retransliuojamos i kartų, o pradinė sistemos apkrova lygi $\rho(0)$. Hong 2010 įrodė, kad sistema išlieka stabili, jei galioja ši sąlyga.

$$\rho(0) < \min \left\{ \frac{(2^{j+1} + 3 \cdot 2^i - i - 4)\sigma T_1 - ((i - 1)2^i + 1)\omega T_1}{i + 1}, (2^{j+1} - 1)\sigma T_1 \right\},$$

kur $1 \leq j \leq 6$.

Norint apkrauti serverį dideliu kiekiu žinučių naudojama *Hping3* programa. *Hping3* – tai (TCP, UDP)/IP paketų generavimo programa. Šis įrankis palaiko tokius protokolus kaip TCP, UDP. *Hping3* įrankio pagrindinės funkcijų sąrašas: TCP ir UDP prievadų stebėjimas; tinklo pralaidumo tikrinimas (naudojant skirtingus protokolus); „*Wireshark*“ programa naudojama sugeneruotų paketų nuskaitymui ir jų išsaugojimui tolimesnei analizei. Eksperimentų procesas automatizuojamas, naudojant sukurtus TCL skriptu, kurie išnaudoja esamų programinių įrankių funkcionalumą. Scenarijai apdoroja nuskaitytų paketų duomenis ir išveda apibendrintus rezultatus.

Hping3 įrankis naudojamas komandinėje eilutėje:

Hping3 [taikiny] [parametrai]

Sugeneruoti srautą reikia parašyti šias komandas 4 lentelė:

4 lentelė.

Srautą generuojančios komandos aprašas

```
for host in {1..254}
do
for host in {1..254}
do
cat invite_tekstas | sed -e s/IPADDRESS/192.168.$host.$host / >stmpl
hping3 -2 -d 680 -E stpmlt -a 192.168.$host.$host -p 5060 192.168.1.123
done
done
```

Tekstas kuris bus siunčiamas pateikiamas 5 lentelėje.\

5 lentelė.

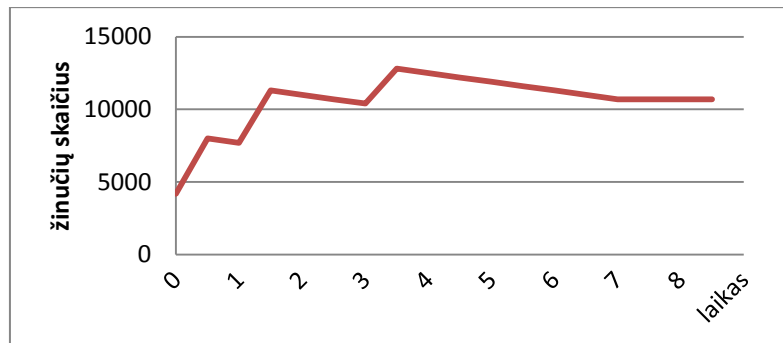
SIP INVITE inicijuojama žinutės aprašas

```
"INVITE sip:ramdzn@192.168.1.123:5060 SIP/2.0\r\n" \
"Via: SIP/2.0/UDP IPADDRESS:5060;branch=z9hG4bK776asdhds\r\n"
"Max-Forwards: 70\r\n" \
"To: ramdzn <sip:ramdzn@test.com>\r\n" \
"From: IPADDRESS<sip:IPADDRESS@test.com>;tag=1928301774\r\n" \
"Call-ID: test-test@ IPADDRESS \r\n" \
"CSeq: 12345 INVITE\r\n" \
>Contact: <sip:ramdzn2@ IPADDRESS:5060>\r\n" \
"Content-Type: application/sdp\r\n" \
"Content-Length: 147\r\n" \
"\r\n" \
"v=0\r\n" \
"o=UserA 2890844526 2890844526 IN IP4 test.com\r\n" \
"s=Session SDP\r\n" \
"c=IN IP4 IPADDRESS \r\n" \
"t=0 0\r\n" \
"m=audio 49172 RTP/AVP 0\r\n" \
"a=rtpmap:0 PCMU/8000" \
"0"
```

Pasinaudodami hping programa sugeneruojame apkrovą patikrinti.

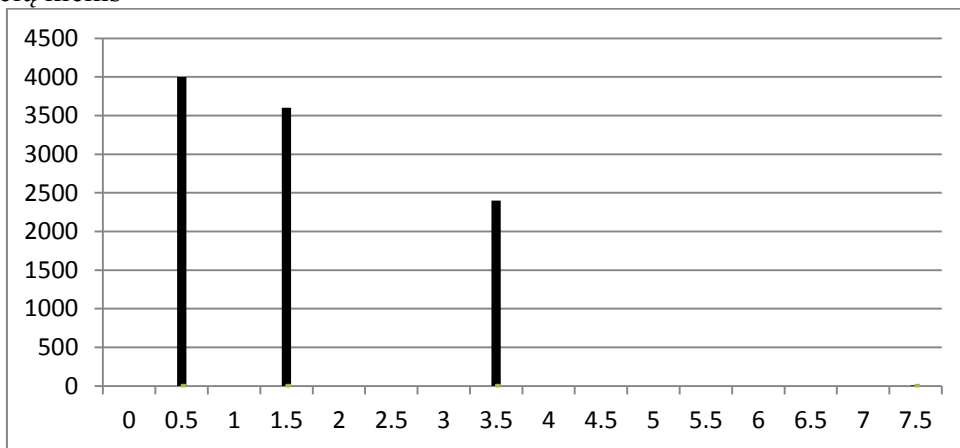
Gauname apdorotų žinučių priklausomybę nuo priimamų žinučių skaičiaus. Pagal šią formulę galime teoriškai numatyti kokio pajėgumo serverio reikės norint apdoroti visas žinutes, nesukeliant serverio ar tinklo perkrovimo.

Modeliuojant realią situaciją, pirminiu momentu, apkrauname serverį pasiūsdami 4200 žinučių. Serveris pajėgus apdoroti 600 žinučių per vieną sekundę. Pasinaudojus (10) formule gauname žinučių eilę pasiekiantį serverį laiko momentu, iki kol apdoros visas žinutes (žr. 5.12 pav.). 5.13 pav. matome retransliuotų žinučių kiekį.



5.12 pav. SIP žinučių kiekio priklausomybė nuo laiko analizės rezultatai eksperimento metu

Žinučių kiekis



Sekundės

5.13 pav. SIP retransliuotų žinučių kiekio priklausomybė nuo laiko

5.6. Penkto Skyriaus išvados

Šiame skyriuje aprašyti technologinės architektūros komponentai, užtikrinantys sistemos komunikacijos veikimą tarp nutolusių heterogeninių

įrenginių. Aprašyti sukonstruoto interpretatorius komponentai, kurie leido integruoti įrenginius į bendrą sistemą duomenų apsikeitimui. Šis pasiūlytas interpretatorius leido atsisakyti daugybės skirtingų protokolų, kurie galimi bendraujant heterogeniniams įrenginiams skirtingo tipo tinkluose. Siūloma eksperimentinė technologinė įranga ir programinė įranga taikant pagrindinį SIP protokolą komunikacijai ir duomenų mainams užtikrinti. Turint vieningą komunikacijos protokolą leidžiama efektyviau (sparčiau) integruoti įrenginius į bendrą sistemą, dėl šios priežasties išvengiame nenumatytų problemų (protokolų nesuderinamumo). Eksperimentiškai tiriama SIP protokolo kritinių žinučių kiekių svyravimai, įrenginių apkrovos, nustatomos sąlygos kaip išvengti tinklo ir serverio perduodamų žinučių perkrovų. Atliktas eksperimentinis tyrimas, kuris patvirtino SIP protokolo naudingumą.

IŠVADOS

1. Atlikta tankių tinklų, lokaliųjų aprėpties tinklų, 3G tinklų, proginių (ad-hoc) tinklų ir lygiarangių (Peer-to-Peer) tinklų galimybių analizė parodė, kad nutolusių įrenginių bendravimui naudojama didelė įvairovė komunikacijos protokolų, o skirtingiems įrenginiams kūjami vis nauji papildomi prietaisai (adapteriai), leidžiantys nutolusiems įrenginiams komunikuoti tarpusavyje esant skirtingiems protokolams. Dėl šios priežasties buvo atlikta SIP galimybių studija. Atlikus sesijos inicijavimo protokolo (SIP) architektūrinių sprendimų analizę, nustatyta, kad įmanoma užtikrinti komunikaciją lygiarangiuose (Peer-to-Peer), 3G ir aukštesnės kartos tinkluose, naudojant vieningą SIP protokolą sesijos užmezgimui, palaikymui ir nutraukimui. Šis technologinis sprendimas leidžia atsisakyti daugybės dabar naudojamų skirtingų protokolų taikomų komunikacijai užtikrinti.
2. Išnagrinėta SIP lokalizacijos paslaugos veikimo architektūra ir pasiūlytos SIP funkcines savybės, t.y., galinio mobilaus įrenginio fiksavimo per IP adresą; kelio tarp tinklo mazgų suradimo galimybes. Pasinaudojus įrenginio fiksacija, taikant SIP adreso žymę, galima dar vienu būdu nustatyti judančio objekto geografinę poziciją ir tai taikyti net ir tuo atveju, kai GPS įrenginys nepriima signalo, pavyzdžiui, kai judantis objektas pasislepia už gamtinių barjerų (tunelyje, tarp kalnų ar pastato viduje).
3. Spalvotaisiais Petri tinklais sudarytas SIP protokolo veikimo modelis leido ištirti atliekamas operacijas ir stebėti imitaciniame modelyje vykdomas būsenas, kiekviename sistemos žingsnyje nustatyti reikiamus realios situacijos vykdymo parametrus, aprašyti tinkamus algoritmus. Spalvotaisiais Petri tinklais sukurti modeliai leido išanalizuoti ryšių kanalais perduodamos informacijos struktūrą, kuri reikalinga keičiant vieną būseną kita, bei matyti, kaip yra atliekami valdančios informacijos perdavimai ir informacijos srautų judėjimai.
4. Pasiūlyta sprendimo paramos sistemos architektūra, kurioje buvo

integruotos išplėtos SIP funkcinės galimybės. Naudojantis pasiūlyta architektūra, buvo išnagrinėta, kaip reikia saugoti judančio objekto sensorių stebėsenos duomenis nutolusiose duomenų saugyklose, pasitelkiant įterptines sensorines sistemas. Ši technologija leido naudoti vieningą duomenų magistralę, kaip integruotą SIP funkciją, kuri leidžia vykdyti duomenų, naudojamų nutolusių judančių objektų stebėsenai ir vertinimui, apsikeitimą. Pasiūlyta sprendimo paramos sistemos architektūra leido apjungti nutolusius skirtingo tipo įrenginius SIP pagrindu, atsisakant papildomų adapterių. Tai suteikė galimybę integruoti naujus judančius objektus į sistemą, nedarant pakeitimų bendroje architektūroje.

5. Pasiūlyta techninė ir programinė prototipinė sistema leido užtikrinti ir išbandyti nutolusių judančių objektų komunikaciją, integruoti ir įvertinti išplėtotas SIP funkcinės savybes bei vieningos duomenų perdavimo magistralės galimybes. Sukurtas eksperimentinių tyrimų stendas realiomis tinklo atlikimo sąlygomis leido apjungti keturių skirtingų tipų įrenginius į visumą. Eksperimento teisingumas buvo patikrintas generuojant žinučių srautus, kurie buvo sėkmingai apdoroti visais keturių tipų nutolusiais įrenginiais.

LITERATŪROS SĄRAŠAS

1. ADEYEYE, M., VENTURA, N., HUMPHREY, D. Mapping third party call control and session handoff in SIP mobility to content sharing and session handoff in the web browsing context. In: Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE. IEEE, 2009. p. 1-6.
2. ACKER, R., et al. Ubiquitous home control based on SIP and presence service. In: Proceedings of the 12th International Conference on Information Integration and Web-based Applications & Services. ACM, 2010. p. 759-762.
3. ANDREASEN, F., BAUGHER, M., WING, D. Session description protocol (SDP) security descriptions for media streams. RFC4568 July, 2006.
4. ANDZIULIS, Arūnas, et al. Priority based tag authentication and routing algorithm for intermodal containers RFID sensor network. Transport, 2012, 27.4: 373-382. Andziulis, A., Plėštys, R., Jakovlev, S., Adomaitis, D., Gerasimov, K., Kurmis, M., Pareigis, V. Priority based tag authentication and routing algorithm for intermodal containers RFID sensor network. Transport, 2012, 27.4: 373-382.
5. Architecture. W3C Working Group Note 11 February 2004. W3C, 2004. Available from:
6. ARPAIA, P., LUCARIELLO, G., ZANESCO, A. Multi-agent remote predictive diagnosis of dangerous good transports. In: Instrumentation and Measurement Technology Conference, 2005. IMTC 2005. Proceedings of the IEEE. IEEE, 2005. p. 1685-1690.
7. ASGHARIAN, Z., et al. A framework for SIP intrusion detection and response systems. In: Computer Networks and Distributed Systems (CNDS), 2011 International Symposium on. IEEE, 2011. p. 100-105.
8. BAI, Y., YE, X., MA, Y. Formal Modeling and Analysis of SIP Using Colored Petri Nets. In: Wireless Communications, Networking and Mobile Computing (WiCOM), 2011 7th International Conference on. IEEE, 2011. p. 1-5.

9. BATARLIENE, Nijole; BAUBLYS, Adolfas. Mobile solutions in road transport. *Transport*, 2007, 22.1: 55-60.
10. BARBOSA, A., et al. Integration of SIP protocol in Android Media Framework. In: *EUROCON-International Conference on Computer as a Tool (EUROCON)*, 2011 IEEE. IEEE, 2011. p. 1-4.
11. BARNAWI, A., et al. Security Analysis and Delay Evaluation for SIP-Based mobile MASS examination system. *International Journal*, 2012, 4.
12. BERGER, A., HEFEEDA, M. Exploiting SIP for botnet communication. In: *Secure Network Protocols, 2009. NPSec 2009. 5th IEEE Workshop on*. IEEE, 2009. p. 31-36.
13. BERTRAN, B., et al. A SIP-based home automation platform: An experimental study. In: *Intelligence in Next Generation Networks, 2009. ICIN 2009. 13th International Conference on*. IEEE, 2009. p. 1-6.
14. Bielskis, A. A., Dzemydienė, D., Denisov, V., Andziulis, A., Drungilas, D. 2009. An approach of multi-agent control of bio robots using intelligent recognition diagnosis of persons with moving disabilities, *Technological and Economic Development of Economy* 15(3): 377–394.
15. Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M. *Web Service*
16. BRAUERS, W., Karel M., et al. Multi objective decision making for road design. *Transport*, 2008, 23.3: 183-193.
17. CAMPI, A., CALLEGATI, F. Calling Procedures in Hybrid SIP Network. In: *Advances in P2P Systems, 2009. AP2PS'09. First International Conference on*. IEEE, 2009. p. 203-208.
18. CHANG, C., TSAI, H. The Design of Video Door Phone and Control System for Home Secure Applications. In: *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on*. IEEE, 2011. p. 1-5.
19. CHEN, E., ITOH, M. A whitelist approach to protect SIP servers from flooding attacks. In: *Communications Quality and Reliability (CQR), 2010 IEEE International Workshop Technical Committee on*. IEEE, 2010. p. 1-6.

20. CHENG, Y., et al. Efficient failover and Load Balancing for dependable SIP proxy servers. In: Computers and Communications, 2008. ISCC 2008. IEEE Symposium on. IEEE, 2008. p. 1153-1158.
21. CHENTOUF, Z. SIP overload control using automatic classification. In: Electronics, Communications and Photonics Conference (SIEPC), 2011 Saudi International. IEEE, 2011. p. 1-6.
22. CHI, C., et al. Performance evaluation of SIP servers. In: Communications and Networking in China, 2008. ChinaCom 2008. Third International Conference on. IEEE, 2008. p. 674-679.
23. CHU, Y., et al. IMS-based Smart Grid System. In: Computer and Information Technology (CIT), 2012 IEEE 12th International Conference on. IEEE, 2012. p. 936-942.
24. CRUZ, T., et al. How to provision and manage off-the-shelf SIP phones in domestic and SOHO environments. In: Local Computer Networks (LCN), 2011 IEEE 36th Conference on. IEEE, 2011. p. 42-49.
25. DAGHER, R., et al. A SIP based P2P architecture for social networking multimedia. In: Proceedings of the 2008 12th IEEE/ACM International Symposium on Distributed Simulation and Real-Time Applications. IEEE Computer Society, 2008. p. 187-193.
26. Ding, L.G. Modelling and Analysis of the INVITE Transaction of the Session Initiation Protocol Using Coloured Petri Nets, in 'the 29th Int. Conf. on Applications and Theory of PetriNets and Other Models of Concurrency', LNCS, vol5062, pp. 132-151, Springer. 2008.
27. DING, Lay G., LIU, L. Modelling and analysis of the INVITE transaction of the session initiation protocol using coloured petri nets. In: Applications and Theory of Petri Nets. Springer Berlin Heidelberg, 2008. p. 132-151.
28. DZEMYDIENĖ, D. Intelektualizuotų informacinių sistemų projektavimas ir taikymas. Vilnius: MRU leidybos centras, 2006.
29. EL-HASSAN, F., et al. A High-Performance Architecture of an XML Processor for SIP-Based Presence. In: Information Technology: New

- Generations, 2009. ITNG'09. Sixth International Conference on. IEEE, 2009. p. 90-95.
30. FABIANO, B., et al. Dangerous good transportation by road: from risk analysis to emergency planning. *Journal of Loss Prevention in the process industries*, 2005, 18.4: 403-413.
 31. FERNG, H., CHRISTANTO, I. A Globally Overlaid Hierarchical P2P-SIP Architecture with Route Optimization. *Parallel and Distributed Systems, IEEE Transactions on*, 2011, 22.11: 1826-1833.
 32. FOKUM, Daniel T., et al. An Open-System Transportation Security Sensor Network: Field-Trial Experiences. *Vehicular Technology, IEEE Transactions on*, 2010, 59.8: 3942-3955.
 33. JARUTIS, A., Gedmantas, R., & Grimaila, V. Investigation of SIP Signaling Messages Servicing Time. // *Electronics and Electrical Engineering*, 2011, 114.8: 35-38.
 34. GARCIA-MARTIN, M. Input 3rd-generation partnership project (3GPP) release 5 requirements on the session initiation protocol (SIP). 2005.
 35. GARROPPO, Rosario G., et al. A prediction-based overload control algorithm for SIP servers. *Network and Service Management, IEEE Transactions on*, 2011, 8.1: 39-51.
 36. GEHLOT, V., HAYRAPETYAN, A. A CPN model of a SIP-based dynamic discovery protocol for webservices in a mobile environment. In: *the 7th Workshop and Tutorial on Practical Use of CPNs and the CPN Tools*, University of Aarhus, Denmark. 2006.
 37. GEHLOT, V., NIGRO, C. Colored Petri Net model of the Session Initiation Protocol (SIP). In: *IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society. IEEE*, 2010. p. 2150-2155.
 38. Ghazisaidi, N., Kassaei, H., Bohlooli, M.S. Integration of WiFi and WiMAX-Mesh Networks. In: *Second International Conference on Advances in Mesh Networks: MESH 2009*, Athens, Glyfada, 2009.

39. GUILLET, T., SERHROUCHNI, A., BADRA, M. Mutual Authentication for SIP: A semantic meaning for the SIP opaque values. In: New Technologies, Mobility and Security, 2008. NTMS'08. IEEE, 2008. p. 1-6.
40. GURBANI, Vijay K., KOLESNIKOV, V. A survey and analysis of media keying techniques in the session initiation protocol (SIP). Communications Surveys & Tutorials, IEEE, 2011, 13.2: 183-198.
41. HANDLEY, M., PERKINS, C., JACOBSON, V. SDP: session description protocol. 2006.
42. HAO, J., ZOU, J., DAI, Y. A real-time payment scheme for SIP service based on hash chain. In: e-Business Engineering, 2008. ICEBE'08. IEEE International Conference on. IEEE, 2008. p. 279-286.
43. HENTEZHARDEH, N., et al. Statistical analysis of self-similar Session Initiation Protocol (SIP) messages for anomaly detection. In: New Technologies, Mobility and Security (NTMS), 2011 4th IFIP International Conference on. IEEE, 2011. p. 1-5.
44. HEO, J., et al. Statistical sip traffic modeling and analysis system. In: Communications and Information Technologies (ISCIT), 2010 International Symposium on. IEEE, 2010. p. 1223-1228.
45. HERRERA, Juan C., et al. Evaluation of Traffic Data Obtained via GPS-Enabled Mobile Phones: the Mobile Century Field Experiment. UC Berkeley: UC Berkeley Center for Future Urban Transport: A Volvo Center of Excellence. 2009.
46. HILT, V., WIDJAJA, I. Controlling overload in networks of SIP servers. In: Network Protocols, 2008. ICNP 2008. IEEE International Conference on. IEEE, 2008. p. 83-93.
47. HLAVACS, H., et al. Babel-SIP: Self-learning SIP message adaptation for increasing SIP-compatibility. In: INFOCOM Workshops 2008, IEEE. IEEE, 2008. p. 1-6.
48. HOMAYOUNI, M., et al. Configuration of a sip signaling network: An experimental analysis. In: INC, IMS and IDC, 2009. NCM'09. Fifth International Joint Conference on. IEEE, 2009. p. 76-81.

49. HOMAYOUNI, M., et al. Overload control in sip servers: Evaluation and improvement. In: Telecommunications (ICT), 2010 IEEE 17th International Conference on. IEEE, 2010. p. 666-672.
50. HONG, Y., HUANG, C., YAN, J.. Design of A PI rate controller for mitigating SIP overload. In: Communications (ICC), 2011 IEEE International Conference on. IEEE, 2011. p. 1-5.
51. HONG, Y., HUANG, C., YAN, J. Stability condition for SIP retransmission mechanism: Analysis and performance evaluation. In: Performance Evaluation of Computer and Telecommunication Systems (SPECTS), 2010 International Symposium on. IEEE, 2010. p. 387-394.
52. HONG, Y., HUANG, C., YAN, J.. Analysis of SIP retransmission probability using a Markov-modulated Poisson process model. In: Network Operations and Management Symposium (NOMS), 2010 IEEE. IEEE, 2010. p. 179-186.
53. HONG-XIA, Z., LIAN-ZHANG, Z. Building dynamic model in UML using colored Petri Nets. In: Computer Network and Multimedia Technology, 2009. CNMT 2009. International Symposium on. IEEE, 2009. p. 1-4.
54. <http://www.w3.org/TR/2004/NOTE-ws-arch-20040211/>.
55. HUANG, C., LEE, C., ZHENG, J. A novel SIP-based route optimization for network mobility. Selected Areas in Communications, IEEE Journal on, 2006, 24.9: 1682-1691.
56. IEEE 802.16-2009. IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed Broadband Wireless Access Systems. The Institute of Electrical and Electronics Engineers, 2009, p. 1–2082.
57. IMRAN, K., JENSEN, T. Performance of parallel signaling between IMS Presence server and Web services. In: Electronic Design, Test and Application (DELTA), 2011 Sixth IEEE International Symposium on. IEEE, 2011. p. 248-253.

58. ISLAM, M., HOQUE, R. SIP over Peer-to-Peer—Implications and existing approaches. In: Computers & Informatics (ISCI), 2011 IEEE Symposium on. IEEE, 2011. p. 261-266.
59. YANG, J., HUANG, F., GOU, S. An optimized algorithm for overload control of SIP signaling network. In: Wireless Communications, Networking and Mobile Computing, 2009. WiCom'09. 5th International Conference on. IEEE, 2009. p. 1-4.
60. YEGANEH, H., et al. Scenarios for testing UPDATE method functionality of SIP user agent in IMS. In: Computer Research and Development (ICCRD), 2011 3rd International Conference on. IEEE, 2011. p. 435-439.
61. YI, Z., CHEN, C. The application of HASH technology in the SIP server. In: Computer Design and Applications (ICCD), 2010 International Conference on. IEEE, 2010. p. V5-278-V5-280.
62. JUNG, J. Y., LEE, J. W. ZigBee device access control and reliable data transmission in ZigBee based health monitoring system. In: Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference on. IEEE, 2008. p. 795-797.
63. KAKLAUSKAS, A., ZAVADSKAS, E. K. Internetinė sprendimų parama. Vilnius: Technika, 2002.
64. KHALLOUF, Z., Trust management in peer-to-peer SIP using the security assertion markup language. In: Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on. IEEE, 2008. p. 1-4.
65. KHOURY, S., JEREZ, N., ABDALLAH, T. Efficient user controlled inter-domain SIP mobility authentication, registration, and call routing. In: Mobile and Ubiquitous Systems: Networking & Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on. IEEE, 2007. p. 1-7.
66. KIM, B., et al. Design and implementation of a ubiquitous ECG monitoring system using SIP and the ZigBee network. In: Future generation communication and networking (fgcn 2007). IEEE, 2007. p. 599-604.

67. KUMAR SINGH, V., SCHULZRINNE, H., BONI, P. A new SIP event package for group membership management in advanced communications. In: Internet Multimedia Services Architecture and Applications, 2008. IMSAA 2008. 2nd International Conference on. IEEE, 2008. p. 1-6.
68. LEE, C., HAN, K., LEE, Y. Efficient Resource Registration and Location Scheme in P2P-SIP, using ID-based Signature. In: Advanced Communication Technology, 2008. ICACT 2008. 10th International Conference on. IEEE, 2008. p. 1823-1827.
69. LEE, D. L., et al. Data management in location-dependent information services. *Pervasive Computing*, IEEE, 2002, 1.3: 65-72.
70. LEU, J., et al. Design and implementation of a low cost DNS-based load balancing solution for the SIP-based VoIP service. In: Asia-Pacific Services Computing Conference, 2008. APSCC'08. IEEE. IEEE, 2008. p. 310-314.
71. LI, H., et al. A rules-based intrusion detection and prevention framework against SIP malformed messages attacks. In: Broadband Network and Multimedia Technology (IC-BNMT), 2010 3rd IEEE International Conference on. IEEE, 2010. p. 700-705.
72. LI, L., et al. Adaptor-based Design and Implementation of Peer-to-Peer SIP. In: Computer Science and Software Engineering, 2008 International Conference on. IEEE, 2008. p. 370-373.
73. LI, L., et al. Locality-aware peer-to-peer SIP. In: Parallel and Distributed Systems, 2008. ICPADS'08. 14th IEEE International Conference on. IEEE, 2008. p. 295-302.
74. LI, W., et al. On Sliding Window Based Change Point Detection for Hybrid SIP DoS Attack. In: Services Computing Conference (APSCC), 2010 IEEE Asia-Pacific. IEEE, 2010. p. 425-432.
75. JANAKIRAMAN, V. S. *DECISION SUPPORT SYSTEMS*, 2008
76. LIANZHANG, Z., CHENGCHANG, R. CPN-based Web Service Composition Technology. In: Information Management, Innovation Management and Industrial Engineering, 2009 International Conference on. IEEE, 2009. p. 114-116.

77. LIAO, J., et al. A demand-driven parsing method for SIP offload in home network. *Consumer Electronics, IEEE Transactions on*, 2009, 55.3: 1308-1314.
78. LIU, L. Uncovering sip vulnerabilities to dos attacks using coloured petri nets. In: *Trust, Security and Privacy in Computing and Communications (TrustCom)*, 2011 IEEE 10th International Conference on. IEEE, 2011. p. 29-36.
79. LIU, L. Verification of the SIP transaction using coloured petri nets. In: *Proceedings of the Thirty-Second Australasian Conference on Computer Science-Volume 91*. Australian Computer Society, Inc., 2009. p. 75-84.
80. LUO, M. PENG, T., LECKIE, C. CPU-based DoS attacks against SIP servers. In: *Network Operations and Management Symposium, 2008. NOMS 2008*. IEEE. IEEE, 2008. p. 41-48.
81. MAHMOUDI, O. BELKHIR, A. Extension of SIP protocol for managing home networks. In: *Programming and Systems (ISPS)*, 2011 10th International Symposium on. IEEE, 2011. p. 80-86.
82. MARKEVIČIUS, V., Navikas, D., Jonynas, V., & Dubauskienė, N. Risk Analysis of SIP Monitoring and Control System User Interface. *Electronics and Electrical Engineering.–Kaunas: Technologija*, 2008, 7: 87
83. MEYER, B., PORTMANN, M. Practical Performance Evaluation of Peer-to-peer Internet Telephony using SIP. In: *Computer and Information Technology Workshops, 2008. CIT Workshops 2008*. IEEE 8th International Conference on. IEEE, 2008. p. 204-209.
84. MICHALSKI, W. Recommendations and Regulations of the European Commission Regarding the Pan-European eCall. *Journal of Telecommunications and Information Technology*, 2009, 4: 2009.
85. OREDOPE, A., PHAM, V., EVANS, B. Deploying IP Multimedia Subsystem (IMS) services in future mobile networks. In: *Communications (NCC)*, 2011 National Conference on. IEEE, 2011. p. 1-5.
86. PETERKIN, R., IONESCU, D., GROZA, V. An architecture for mobile sensor network control using ims and reconfigurable hardware. In: *Electronic*

- Design, Test and Application (DELTA), 2011 Sixth IEEE International Symposium on. IEEE, 2011. p. 269-274.
87. PRANEVICIUS, H., Paulauskaite-Taraseviciene, A., & Jarutis, A. Simulation of protocol for initiation of communication sessions using dynPLA mode. *I. Electronics and Electrical Engineering.–Kaunas: Technologija*, 2010, 5: 101.
88. PLESTYS, R.; ZAKAREVICIUS, R. Request and response zone control for routing in MANET. In: *Electronics Conference (BEC), 2010 12th Biennial Baltic. IEEE*, 2010. p. 219-222.
89. PLESTYS, Rimantas; ZAKAREVICIUS, Rokas. Variable Response Zone Routing for Ad Hoc Networks. *Information Technologies*, 2009, 158-164.
90. QIN, S., et al. Design and implementation of SIP User-Agent based on applet. In: *Broadband Network and Multimedia Technology (IC-BNMT), 2010 3rd IEEE International Conference on. IEEE*, 2010. p. 780-784.
91. QIULIN, T., et al. Design of a latticing min-type system with monitoring and tracking localized for transporting hazardous chemicals. In: *Computer Application and System Modeling (ICCASM), 2010 International Conference on. IEEE*, 2010. p. V13-391-V13-394.
92. RAFIQUE, Zubair, M., ALI AKBAR, M., FAROOQ, M. Evaluating DoS attacks against SIP-based VoIP systems. In: *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE. IEEE*, 2009. p. 1-6.
93. ROLY, A., SCHUMACHER, L. SIP overload control testbed: Design, building and validation tests. In: *Consumer Communications and Networking Conference (CCNC), 2011 IEEE. IEEE*, 2011. p. 228-230.
94. ROSENBERG, J., et al. SIP: session initiation protocol. RFC 3261, Internet Engineering Task Force, 2002.
95. RUIFANG, M. Environmental risk assessment model on dangerous goods during transportation. In: *Supply Chain Management and Information Systems (SCMIS), 2010 8th International Conference on. IEEE*, 2010. p. 1-5.

96. RUSINOVIC, Z., BOGUNOVIC, N. Self-healing model for SIP-based services. In: Telecommunications, 2009. ConTEL 2009. 10th International Conference on. IEEE, 2009. p. 375-379.
97. RIPEANU, Matei. Peer-to-peer architecture case study: Gnutella network. In:Peer-to-Peer Computing, 2001. Proceedings. First International Conference on. IEEE, 2001. p. 99-100.
98. SARIKAYA, B., ZHENG, X. SIP paging and tracking of wireless LAN hosts for VoIP. IEEE/ACM Transactions on Networking (TON), 2008, 16.3: 539-548.
99. SCHMIDT, H., et al. How well does JXTA fit Peer-to-Peer SIP?. In:Communications, 2008. ICC'08. IEEE International Conference on. IEEE, 2008. p. 1792-1796.
100. SCHOR, L.,SOMMER, P.,WATTENHOFER, R. Towards a zero-configuration wireless sensor network architecture for smart buildings. In: Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings. ACM, 2009. p. 31-36.
101. SHAN, L., JIANG, N. Research on security mechanisms of SIP-based VoIP system. In: Hybrid Intelligent Systems, 2009. HIS'09. Ninth International Conference on. IEEE, 2009. p. 408-410.
102. SHIM, E., KRISHNASWAMY, V., SCHULZRINNE, H. Automatic phone number mapping verification for phone number based SIP peering. In: Consumer Communications and Networking Conference (CCNC), 2011 IEEE. IEEE, 2011. p. 995-999.
103. SISALEM, D., FLOROIU, J., TEKELEC, T. Protecting VoIP services against DoS using overload control. In: Proceedings of The 13th Nordic Workshop on Secure IT Systems NordSec 2008. 2008. p. 147.
104. SUBRAMANIAN, Sureshkumar, V., DUTTA, R. Comparative study of secure vs. Non-secure transport protocols on the SIP proxy server performance: An experimental approach. In: Advances in Recent Technologies in Communication and Computing (ARTCom), 2010 International Conference on. IEEE, 2010. p. 301-305.

105. STOICA, Ion, et al. Chord: A scalable peer-to-peer lookup service for internet applications. In: ACM SIGCOMM Computer Communication Review. ACM, 2001. p. 149-160.
106. SUN, J., et al. Flow management for SIP application servers. In: Communications, 2007. ICC'07. IEEE International Conference on. IEEE, 2007. p. 646-652.
107. TANNA, P. SIP: Ringing timer support for INVITE Client Transaction. In: Computing & Informatics, 2006. ICOCI'06. International Conference on. IEEE, 2006. p. 1-6.
108. TOUCEDA, D. S., et al. Advantages of identity certificate segregation in P2PSIP systems. IET communications, 2011, 5.6: 879-889.
109. TSENG, Y., CHEN, J., CHENG, Y. Design and implementation of a SIP-based mobile and vehicular wireless network with push mechanism. Vehicular Technology, IEEE Transactions on, 2007, 56.6: 3408-3420.
110. USUI, T., et al. On locating loss links of signaling messages in SIP-based services. In: Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE. IEEE, 2009. p. 1-7.
111. VALENTE, F., et al. A telecommunications framework for real-time monitoring of dangerous goods transport. In: Intelligent Transport Systems Telecommunications,(ITST), 2009 9th International Conference on. IEEE, 2009. p. 13-18.
112. WAN, X., LI, Z., FAN, Z. A SIP DoS flooding attack defense mechanism based on priority class queue. In: Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on. IEEE, 2010. p. 428-431.
113. WU, B. An Extensive Scheme for SIP-Based Mobile Network Fast Handoff. In: Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on. IEEE, 2010. p. 1-4.
114. WU, W., et al. A Fast Failure Detection and Failover Scheme for SIP High Availability Networks. In: Dependable Computing, 2007. PRDC 2007. 13th Pacific Rim International Symposium on. IEEE, 2007. p. 187-190.

115. XIE, C., LUO, J., GUO, L. Estimating radioactive material release risks in transport accidents. In: Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE), 2011 International Conference on. IEEE, 2011. p. 994-997.
116. ZHANG, L., MIYAJIMA, H., HAYASHI, H. An effective SIP security solution for heterogeneous mobile networks. In: Communications, 2009. ICC'09. IEEE International Conference on. IEEE, 2009. p. 1-5.
117. ZHANG, X., LEI, W., ZHANG, W. Using P2P network to transmit media stream in SIP-based system. In: Young Computer Scientists, 2008. ICYCS 2008. The 9th International Conference for. IEEE, 2008. p. 362-367.
118. ZHOU, J., WU, T., LENG, J. Research on voice codec algorithms of SIP phone based on embedded system. In: Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on. IEEE, 2010. p. 183-187.
119. ZHU, L., SHAN, X. Colored Petri net based workflow model mapping. In: Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference on. IEEE, 2010. p. 2743-2747.
120. ZOU, J., DAI, Y. Motivating and modeling SIP offload. In: Computer Communications and Networks, 2007. ICCCN 2007. Proceedings of 16th International Conference on. IEEE, 2007. p. 741-746.
121. Xlite. <http://www.counterpath.com/x-lite.html>. 2013
122. EKIGA. <http://www.ekiga.org/>, 2013
123. SKYPE. www.skype.com. 2013

Autoriaus publikacijų disertacijos tema sąrašas

MOKSLO STRAIPSNIAI

Straipsniai mokslo žurnaluose, referuojamuose ir turinčiuose citavimo indeksą Mokslinės informacijos instituto duomenų bazėje „ISI Web of Science“

1. Dzemydienė, D.; Dzindzalieta, R. Development of Architecture of Embedded Decision Support Systems for Risk Evaluation of Transportation of Dangerous Goods// *Technological and Economic Development of Economy*. ISSN 1392-8619. 2010, 6(4): 654–671. IF=5.605.

Straipsniai mokslo žurnaluose, referuojamuose kitose tarptautinėse duomenų bazėse

2. Dzemydienė, D., Dzindzalieta, R., Andziulis, A., Krylovas, A., Rudzkienė, V. Componential risk evaluation of transportation of dangerous goods by means of monitoring and localization// *International Journal of Applied Science & Computations* / Editor S.K. Dey. USA. ISSN 1089-0025, Vol. 18, No. 1, May, 2011. p. 41-50. (Math Sci Index).
3. Dzemydienė, D., Dzindzalieta, R. Mobilijų objektų komunikacijos modeliavimas ir valdymas taikant spalvotuosius Petri tinklus// *Socialinės technologijos: mokslo darbai = Social technologies: research papers / Mykolo Romerio universitetas*. ISSN 2029-7564 2011, [Nr.] 1(1). p. 193–204. (Copernicus, EBSCO)
4. Dzemydienė D., Dzindzalieta R. Mobilijų technologijų taikymas judančių transporto objektų stebėsenai ir komunikavimui // *Informacijos Mokslai*. Vilniaus universiteto leidykla, ISSN 1392-0561, 2009: 274-280. (CEOL, Copernicus)

Moksliniai straipsniai recenzuojamuose konferencijų pranešimų leidiniuose

5. Dzemydienė, D.; Dzindzalieta, R. Multi-layered architecture of decision support system for monitoring of dangerous good transportation // In: *Databases and information systems: Tenth International Baltic Conference on Databases and Information systems: Local Proceedings, Materials of Doctoral Concorcium*. July 8-11, 2012, Lithuania / (Eds) Albertas Čaplinskas,

- Dzemyda, G., Lupeikienė, A., Vasilecas, O.; Lithuanian Academy of Sciences, Vilnius University, Lithuanian Computer Society. Vilnius: Žara, ISBN 9789986342748. 2012. Vol. 924, p. 128-141.
6. Dzindzalieta, R. SIP Protocol as a Communication Bus to Control Embedded Devices// In: Databases and information systems: Tenth International Baltic Conference on Databases and Information systems: Local Proceedings, Materials of Doctoral Consortium. July 8-11, 2012, Lithuania / (Eds) Albertas Čaplinskas, Dzemyda, G., Lupeikienė, A., Vasilecas, O.; Lithuanian Academy of Sciences, Vilnius University, Lithuanian Computer Society. Vilnius: Žara, ISBN 9789986342748. 2012. Vol. 924, 229-234.
 7. Dzemydienė, Dalė, Dzindzalieta, Ramūnas, Perspectives of session initialization protocol as universal communication bus in mobile networks. // Social technologies '11. ICT for Social Transformations: Conference Proceedings, Vilnius-net, November 17-18, 2011 [Elektroninis išteklius]. Vilnius: Mykolo Romerio universitetas, ISBN 9789955193784, 2011, p. 104-105.
 8. Dzemydienė, D., Dzindzalieta, R. Development of Decision Support System for Risk Evaluation of Transportation of Dangerous Goods Using Mobile Technologies. 5th International Vilnius conference and EURO-mini conference "Knowledge-based technologies and OR methodologies for decisions of sustainable development", (Eds.) M.Grasserbauer, L.Sakalauskas, E.K.Zavadskas. Vilnius : Technika, 2009, p. 108-113.
 9. Dzemydiene, D.; Bielskis, A.A.; Andziulis, A.; Drungilas, D.; Dzindzalieta, R.; Gricius, G. The Reinforcement Framework of Decision Support System for Localization and Monitoring of Intelligent Remote Bio Robots // In (Eds.) Igor V. Kabashkin, Irina V. Yatskiv, Proceedings the 10th International Conference „Reliability and Statistics in Transportation and Communication“ (RelStat’10), 20-23 October 2010. Riga, Latvia. ISBN 9789. 2010, p. 207-217.

Kituose periodiniuose recenzuojamuose leidiniuose

10. Dzemydienė, D.; Dzindzalieta, R.; Bielskis, A. A.; Andziulis, A.; Drungilas, D. Sensorinių tinklų taikymo pavyzdžiai intelektualiai aplinkai kurti belaidžių technologijų priemonėmis. // *Technologijos mokslo darbai Vakarų Lietuvoje* / Klaipėdos universiteto Leidykla. ISSN 1822-4652. 2010, [D.] 7, p. 160-164.

Mokomosios metodinės priemonės

11. Norvaišas, S., Dzemydienė, D., Dzindzalieta, R., Kalinauskas, M., Okulič-Kazarinas, M., Naujikienė, R., Šiugždaitė, R. Įtinkintos informatikos studijos: metodinė priemonė / Recenzavo: Irmantas Rotomskis, Jūratė Skūpienė. Vilnius: Mykolo Romerio universiteto Leidybos centras, 2011. 117 p. : iliustr., lent. ISBN 9789955192817.

Tezės

12. Dzindzalieta, R., Dzemydienė, D. Development of architecture for mobile service system for location of moving objects using mobile technologies // Operacijų tyrimai verslui ir socialiniams procesams: 3-ioji Lietuvos jaunujų mokslininkų konferencija: programa ir tezės, 2010 m. spalio mėn. 1 d. / Lietuvos operacijų tyrimų draugija, Matematikos ir informatikos institutas, Mykolo Romerio universitetas, Vilniaus Gedimino technikos universitetas. Vilnius : [Mykolo Romerio universitetas], 2010, p. 10.