

TALLINNA TEHNIKAÜLIKOOL

Raadio- ja sidetehnika instituut

Kood: IRT84LT

Multimeedia kasutus piiratud sidekanalites

Marika Kulmar

Töö on tehtud ...telekommunikatsiooni..... õppetooli juures

Juhendaja ...Avo Ots.....

Kaitsmine toimub raadio- ja sidetehnika instituudi kaitsmiskomisjonis

Autor taotleb tehnikateaduste magistrikraadi

Esitatud: 26.04.2002.

Kaitsmine:

Tallinn 2002

Referaat

Töö vaatlleb helist ja videopildist koosneva multimeediasignaali arvutivõrgu kaudu kuulajale-vaatajale edastamist. Töö esimese osas tutvustatakse lühidalt riistvara ja tarkvara näiteks loengute interneti kaudu levitamiseks. Töö teises osas uuritakse, kuidas mõjutavad erinevad võrgu parameetrid (latentsus, läbilaskevõime, paketikadu ja bitivea tõenäosus) multimeedia andmevoo edastamist kasutajale. Samuti uuritakse, milline transpordiprotokoll sobib paremini reaalaja andmevoogu edastama ja tuuakse näide ühest levinud multimeedia terminali tarkvarast (RealPlayer).

Töös antakse soovitusi multimeedia terminali tarkvara valikuks ning seadistamiseks kasutaja kasutuslihtsusest lähtudes.

Töös näidatakse, et enim halveneb andmevoo edastuse kvaliteet juhul kui võrgu allalüli läbilaskevõime on väiksem multimeedia andmevoo edastuskiirusest. Samuti on suure mõjuga parameeter võrgu latentsus. RealPlayeri poolt kasutatavatest transpordiprotokollidest oli kõige parem UDP.

Töös toodud näidete põhjal võib väita, et multimeedia andmevoog, mille edastuskiirus on 20 kb/s, on mobiilandmesidevõrgu GPRS kaudu kvaliteetselt kuulatav-vaadatav, kui võrk on vähe koormatud.

Võtmesõnad on multimeedia, audio, video, videokonverents, edastuskiirus, latentsus, läbilaskevõime, paketikadu, bitivea tõenäosus, GPRS.

Töö on kirjutatud eesti keeles, koosneb 90 leheküljest tekstist 80 joonisest ja 3 tabelist.

Summary

This master thesis deals with transmitting multimedia consisting audio and video signals using computer network to users. The first part of thesis introduces hardware, software and methods for digitalizing analog audio and video signals and broadcasting that digitalized multimedia via Internet. In the second part of thesis it is studied how do different network parameters (latency, data rate, packet loss and bit error rate) affect multimedia streaming quality and which transport protocol is most suitable for this task. An example of widely used software RealPlayer is tested.

Recommendations for choosing multimedia terminal software and its configuration are given at a point of user's simplicity to use this software.

It is showed that biggest affect to datastream transmission quality has downlink data rate, if it is less than encoding rate of clip. Second is latency. Best transport protocol to use is UDP.

We can claim, that multimedia datastream with encoding rate 20 kbps has good quality when transmitted in GPRS network with less network load according to the test results.

Keywords are multimedia, audio, video, videoconference, bandwidth, latency, data rate, packet loss, bit error rate, GPRS.

Thesis is written in Estonian and it consists of 90 pages of text, 80 figures and 3 tables.

Eessõna

Käesolev magistritöö käsitleb multimeedia digitaliseerimist ja selle pakettside võrgus edastamist. Lisaks paljudele muudele eesmärkidele saab seda kasutada ka loengute videosalvestuste interneti kaudu levitamiseks. Selline vajadus ja idee tekkisid Side loengu juures, kui üliõpilasi oli nii palju, et auditoorium jäi väikeseks.

Mobiilside võrkudes võetakse kasutusele tehnoloogiaid, mis võimaldavad pääsu internetti, siis on ka vaja uurida, kas ja kuidas praegused arvutivõrgu rakendused mobiilvõrgus ka töötavad. On võetud väike lõik Side loengu videost, mille kohta eeldame, et seda võiks mobiilvõrgu kaudu vaadata, ja testitakse, kuidas selle vaatamisele võrguparameetrid mõjuvad.

Internetile pääsetakse ligi enamasti veebilehe kaudu ja kõik vajalikud rakendused võiks kasutajale sealtkaudu ligipääsetavad olla. Sisu juurde pääsemine peaks olema võimalikult lihtne ja mugav ning mitte vajada halvasti kättesaadavat ja keeruliselt kasutatavat tarkvara.

Magistritöö valmimisele aitasid kaasa hr Avo Ots, kes tekitas Side loengu tegemise probleemi lahendamise vajaduse, ja MAI keskus, kus on olemas mobiilse pakettvõrgu simulaatorid ja minu töökoht. Siinkohal tänan Avo Otsa, kes magistritöö tegemisel andis vabad käed.

Sisukord

Sissejuhatus	8
1 Multimeedia digitaliseerimine	10
1.1 Multimeedia taju iseärasus	10
1.2 Heli digitaliseerimine	10
1.2.1 Heli koodekid	11
1.3 Video digitaliseerimine	12
1.3.1 Värvimudelid	13
1.3.2 Pakkimise meetodid	14
1.3.3 Video koodekid ja algoritmid	15
1.4 Multimeedia edastamine	16
1.4.1 Edastuse transpordi protokollid	17
1.4.2 Multisaade	20
1.4.3 Videokonverents	21
1.4.4 Multimeedia andmevoogude eriomadused ja nõuded võrgule	22
1.5 Multimeedia riistvara	23
1.5.1 Audioseadmed	23
1.5.2 Videoseadmed	23
1.6 Tarkvara	24
1.6.1 Microsoft Netmeeting	24
1.6.2 Mbone Tools	25
1.6.3 Real Networks tarkvara	31
1.7 Teiste kogemused	33
1.7.1 Kalifornia ülikoolis toimuvad loengud	33
1.7.2 BIBS	37
1.7.3 Lecture Browser	38
1.7.4 Kogemused Eestis	38
1.8 Kokkuvõte	38
2 Multimeedia ülekanne piiratud ressursside puhul	40
2.1 Kirjeldus	40
2.1.1 Riistvara	40
2.1.2 Tarkvara	40
2.1.3 Seaded	41
2.1.4 Töö käik	42
2.1.5 Kogutavad andmed	42
2.2 Andmevoo profiil piiranguteta võrguga	43
2.2.1 Andmevoo edastus	43
2.3 Interpreteeritud andmed	44
2.3.1 Piiramata võrgutingimuste korral saadud testitulemused	44
2.3.2 Ettemängimise kvaliteet	46
2.3.3 Aeg	47
2.3.4 Andmehulk	56
2.3.5 Pakettide arv	65
2.3.6 Paketi pikkus	72
2.3.7 RealPlayeri edastuskiiruse statistika	75
2.3.8 RealPlayer pakettide arv	78
2.3.9 Pakettide arvu võrdlus	81
2.3.10 Katsete kokkuvõte	84
2.4 Majanduslik külg	85
Kokkuvõte	88

Tähistuste loetelu

ADPCM – *Adaptive Differential Pulse Code Modulation* – adaptiivne diferentsiaalne impulssmodulatsioon

BIBS – *Berkeley Internet Broadcasting System*

BMRC – *Berkeley Multimedia Research Center*

CD – *compact disc* – laserplaat

CELP – *Codebook Excited Linear prediction* – lineaarse ennustamisega helikoodek.

CIF – *Common Intermediate Format* – videokonverentsistandard

CMYK – *Cyan, Magenta, Yellow, black* – värvimudel värviliste kujutiste printimiseks

DCT – *Discrete Cosine Transform* – diskreetne koosinusteisendus

DVD-ROM – *Digital Versatile Disk read-only memory* – “universaaldigitaalketas”, laserkettastandard

DVI – *digital video interactive* – videofilmi salvestuse tehnoloogia

EPAN – *Ethernet Protocol Analyzer* – arvutivõrgu protokollide analüsaator, programm

GPRS – *General Packet Radio Service* – üldine raadio-pakettandmesideteenus

HTTP – *Hypertext Transfer Protocol* – klient-server-protokoll dokumentide edastuseks veebis

IP – *Internet Protocol* – võrgukihi protokoll

ISDN – *integrated services digital network* – integraalteenuste digitaalvõrk

ITU-T – *International Telecommunication Union Telecommunication Standardization Sector* – Rahvusvahelise Elekterside Liidu Sidestandardisektor

JPEG – *Joint Photographic Experts Group* – “fotospetsialistide koondrühm”, standardne värvi- ja pooltoonpiltide pakkimise algoritm

LAN – *Local Area Network* – kohtvõrk

LPC – *Linear Prediction Coding* – lineaarse ennustamisega kodeerimine

MATE – *Mobile Application Test Environment* – mobiilirakenduste testkeskkond

MIME – *Multipurpose Internet Mail Extensions* – multimeil, võimaldab edastada mittetekstilisi andmeid

MPEG – *Motion Picture Experts Group* – heli- ja videoteabe pakkimise standard

MP3 – *MPEG Audio Layer-3* – heli pakkimise standard

M-JPEG – *Motion JPEG* – videofilmi salvestusstandard

NTSC – *National TV Standards Committee* – värviteleviseiooni standard, kasutuses Ameerika Ühendriikides

OSI – *Open System Interconnection Architecture* – avatud süsteemide ühendamise arhitektuur

PAL – *Phase Alternating Line* – videosignaali standard

PCM – *Pulse Code Modulation* – impulssmodulatsioon

PLAY – *play* – ettemängimise käsk RealPlayeris

QCIF – *Quarter Common Intermediate Format* – videokonverentsi standard

RFC – *Request For Comments* – kommentaarikutse Interneti protokollistikku ja sellega seotud katseid kirjeldav dokumendisari

RGB – *red, green, blue* – värvuste teisendus nende kuvamiseks arvutiekraanil

RLE – *run-length encoding* – sümbolite jada kodeerimine, video pakkimise algoritm

RTCP – *RTP Control Protocol* – Reaalaja transpordi juhtprotokoll

RTP – *Real-time Transport Protocol* – Reaalaja transpordi protokoll

RTSP – *Real-time Streaming Protocol* – Reaalaja andmevoo protokoll

SQCIF – *Sub Quarter Common Intermediate Format* – videokonverentsi standard

SECAM – *Systeme En Couleur Avec Memoire* – värviteleviseiooni standard, loodud Prantsusmaal

TCP – *Transmission Control Protocol* – edastusohje protokoll

TTL – *Time To Live* – paketi eluiga

TTÜ – Tallinna Tehnikaülikool

UDP – *User Datagram Protocol* – datagrammiprotokoll

UMTP – *UDP Multicast Tunneling Protocol* – multisaate tunneli protokoll

USB – *universal serial bus* – universaalse aeglase seadmete jadasiini standard

XDSL – *Digital Subscriber Line* – digitaalne abonendiliin, X tähendab ükskõik millist sellist tehnoloogiat

YUV – video värvimudel

Sissejuhatus

Digitaalne multimeedia leiab järjest laialdasemat kasutust aina vähem arvutitundvate internetikasutajate juures. Multimeedia loomiseks on kasutusel seadmed, mis tekitavad analoogsignaale. Andmevõrkudes saab edastada ainult digitaalseid andmeid. Multimeedia digitaliseerimiseks on olemas riistvara ja tarkvara, mis täieneb kogu aeg, võimaldamaks multimeedia edastamist aina väiksema läbilaskevõimega võrkudes.

Tänapäeval, kui arvutivõrgud võimaldavad edastada väga suuri andmehulki, on nad aina sobivamad suuremahulise multimeedia edastamiseks. Sama areng toimub ka mobiilvõrkudes. Multimeedia saab andmemahukuselt suurimaks edastatavaks sisuks internetis. Võrgud ja rakendused on nüüd tarbija unistustele järele jõudnud. Videokonverentsid ja internetist videote vaatamised (mitte allalaadimised) on populaarseks saanud.

Multimeediat saab ära kasutada ka koolituseks. Loengud saab salvestada videosse ja neid levitada videona. Siin tuleb levitamisel appi “kõikide võrkude võrk” internet, millel pole piire. Internetis videote levitamiseks on videod vaja viia sobivasse digitaalsesse vormingusse ja see vorming oleneb ka sellest, milleks seda videot kasutatakse. Analoogsignaali digitaliseerimisel on võimalik saada väga suuremahulisi digitaalseid videosid, aga kas seda suurt mahtu ikka on vaja, kui videot vaadatakse väikses aknas arvutimonitorilt? Siin tulevad appi pakkimise meetodid, millega on võimalik videost teha täpselt vajaliku kvaliteedi ja andmemahuga digitaalne klipp.

Nende videote vaatamine on samuti mugavamaks läinud. Enam ei pea tervet faili alla laadima, et vaadata korraks selle sisse. Andmevoo edastamise tehnoloogia korral edastatakse serverist väike osa videofaili algusest ja ülejäänud osa edastatakse faili vaatamise ajal.

Telefon ja arvuti aina lähenevad üksteisele. Varem olid telefoni ja arvutivõrk eraldi, nüüd lähenevad nad üksteisele aina rohkem, kuni interneti protokollid kasutatakse mõlemates. Telefoni- ja mobiiltelefonivõrk on esialgu veel üsna väikse läbilaskevõimega, aga siin mobiilside kolmas põlvkond teeb oma töö ja varsti pole vahet, millise võrgu kaudu internetti kasutatakse.

Ka mobiilses andmesidevõrgus saab võimalikuks videosid edastada. See hakkab toimuma sõnumiedastuse põhimõttel. Mobiilside kolmanda põlvkonna võrkude läbilaskevõime saab sobivaks videote levitamiseks.

Teenuse populaarsuse suurenedes on märgata selle teenuse hinna odavnemist. Interneti kasutamise maksustamises on hakanud uued tuuled puhuma mobiilvõrgu poolelt, kus andmesidet maksustatakse ajakasutuse asemel edastatud andmehulga eest. Olenevalt interneti rakenduse andmemahukusest võib veeta internetis tunde ja maksta andmehulga põhiselt sama palju kui ajapõhise maksustamise korral mõne minuti internetis veedetud aja eest.

Veebileht avab pääsu info juurde, olgu see info ükskõik millisel kujul – tekst, pilt, video. Infot esitatakse vastavalt kasutaja seadme – ükskõik, kas arvuti või telefoni – võimalustele ja andmed on erinevas formaadis. Kasutaja ei pea ise midagi erilist tegema, vaid hiirega sobivale lingile klõpsima, server saadab tema terminalile õiges formaadis andmed näitamiseks. Kasutaja ligipääs videotele olgu võimalikult lihtne. Selleks on kõige parem tavaline veebileht ja kasutaja poolt sooritatav väike hiireklõpsude arv. Kasutaja ei pea videole ligipääsuks ümber õppima.

Käesoleva magistritöö ülesandeks on vaadelda meetodeid ja tarkvara, kuidas multimeediat edastada internetis otseülekanadena. Praktiline vajadus on TTÜs sügissemestril õpetatava aine Side õpetamine, kus loengu kuulajaid on nii palju, et nad ei mahu ära ühte auditooriumisse. Loengus kasutatavate auditooriumite vahel on võimalik luua arvutivõrgu ja interneti ühendus. Ülesanne laieneb edasi, et kuulajad asuvad lisaks neile auditooriumitele ka internetis.

Loengute internetti viimiseks on mitut tüüpi tarkvara: nii sellist, kus interneti kaudu vaatavad saavad ka oma sõna sekka öelda, aga neil võib olla võimalus oma mõtteid teksti või joonise kujul edastada loengu pidajale loengu ajal – nn interaktiivne, kui ka sellist, kus vaataja ei saa loengu ajal oma sõna sekka öelda. Loengus kasutatavate abimaterjalide sidumiseks loenguga on jällegi internet kui põhjatu infoallikas abiks. Loengu ajal saavad kuulajad, kes niigi interneti kaudu loengu videot vaatavad, vaadata ka selliseid veebilehti, kus need abimaterjalid asuvad. Alati ei peagi selleks uut brauseriakent avama, vaid videot näitav tarkvara võib selle töö ise teha. On tarkvara, mis sünkroniseerib üksteisega nii videoklipi audio ja video osa kui ka pildid, veebilehed ja muud programmid, mida loengu pidaja vajalikuks peab näidata. Need materjalid ei pea olema kuskil internetis, vaid võivad olla ka loenguvideo osad. Kasutajal jääb üle vaid jälgida mängu.

Töö teise osa eesmärk on uurida, kuidas toimub mobiilsidevõrgus loenguvideo edastamine ja millised võrguparameetrid mõjutavad selle kvaliteeti. Mobiilsidevõrkudest on teada, et läbilaskevõime on väike, latentsus suur ja võimalikud võrgukatkestused ning üleslüli ja allalüli on erinevate parameetritega. Testimiseks koostati videoklipp, mis on kodeeritud kõige väiksema edastuskiirusega 20 kb/s, mis selle tarkvaraga võimalik oli. See kodeeritud edastuskiirus vastab GPRS võrgus saavutatud läbilaskevõimele 26-53 kb/s. Eeldatavalt on seda klippi võimalik edukalt edastada GPRS võrgu kaudu, kuna edastuskiirus on väiksem kui võrgu läbilaskevõime. Sellise oletuse juures jääb testimisel vaadelda, kuidas GPRS võrgu teised teadaolevad omadused edastuse kvaliteeti mõjutavad.

1 Multimeedia digitaliseerimine

Selles peatükis vaatleme eraldi heli ja video digitaliseerimist, kodeerimist, pakkimist, edastamist ja ettemängimist.

Analoogkujul heli- ja videosignaalid tuleks arvutis kasutamiseks viia digitaalsele kujule. Selleks need signaalid diskreeditakse piisava diskreetimissagedusega ja diskreetide väärtused salvestatakse piisavalt pikkadesse andmeväljadesse. Olenevalt selle multimeedia kasutuseesmärgist on vaja erinevat heli ja video kvaliteeti. Pakkimata digitaliseeritud multimeedia andmed hõivavad väga suure salvestusmahu. Multimeedia andmete pakkimisel arvestatakse ka lisaks inimese taju iseärasusi, sh millised on vähimad võimalikud muutused (katkestused) helis ja pildis, mida inimene veel tajub. Kui heli ja videot töödeldakse eraldi, siis peab samaaegse ettemängimise tarvis mõlemad andmevood sünkroniseerima.

1.1 Multimeedia taju iseärasus

Multimeedia taju iseärasus on hästi häälestatud audio ja video sünkroonsusele st esitatavas multimeedias peavad inimese huuled liikuma sünkroonselt tema häälega.

Enamus heli ja video rakendusi vajavad täpset takti, mille järgi andmed eemaldatakse puhvrilt ja mängitakse ette. Heli on väga tundlik ettemängimise taktile, sest inimkõrv tajub ettemängimise takti muutust muutusena heli kõrguses. [17] Suuremad viite muutused kui 20-40 ms põhjustavad tajutavat katkendlikkust videos. Enamus audio ja video digitaliseerimise algoritme ja edastuse protokolle on tundlikud viite muutumisele. Andmevooga (*streaming*) audio ja video edastusprotokollid kasutavad edastusvõrgus tekkiva viite muutumise kompenseerimiseks väikest ettemängimise puhvrit. Kui multimeedia rakendusele saabub andmeid ettemängimiseks liiga palju või liiga vähe, siis tekib vastavalt puhvri ületäitumine või puhvri tühjenemise järel andmete puudujääk, mille tõttu ettemängimise kvaliteet halveneb.

Moodsate video ja audio digitaliseerimise algoritmide kasutamisel ülekandel tekkinud vähesed bitivead pole multimeedia ettemängimisel inimesele tajutavad. [17]

1.2 Heli digitaliseerimine

Käesolevas töös eelistame multimeedias heli. Kui mingil põhjusel pole võimalik kõiki multimeedias vajaminevaid andmeid kas edastada või ette mängida, siis heli jaoks tuleb teha kõik võimalik ja videopildi pealt võib ressursse kokku hoida. Meie ülesandes, loengus, on õppejõu jutt tähtsam kui õppejõu kujutis ja loengumaterjalid on üliõpilastele kättesaadavad ka failidena.

Diskreetimissagedus peab olema vähemalt 2 korda suurem kui sisendsignaali suurim sagedus. Olenevalt heli iseloomust diskreeditakse seda erinevate diskreetimissagedustega. Inimese hääle jaoks kasutatakse 8 kHz, muusika jaoks kuni 44,1 kHz. [7]

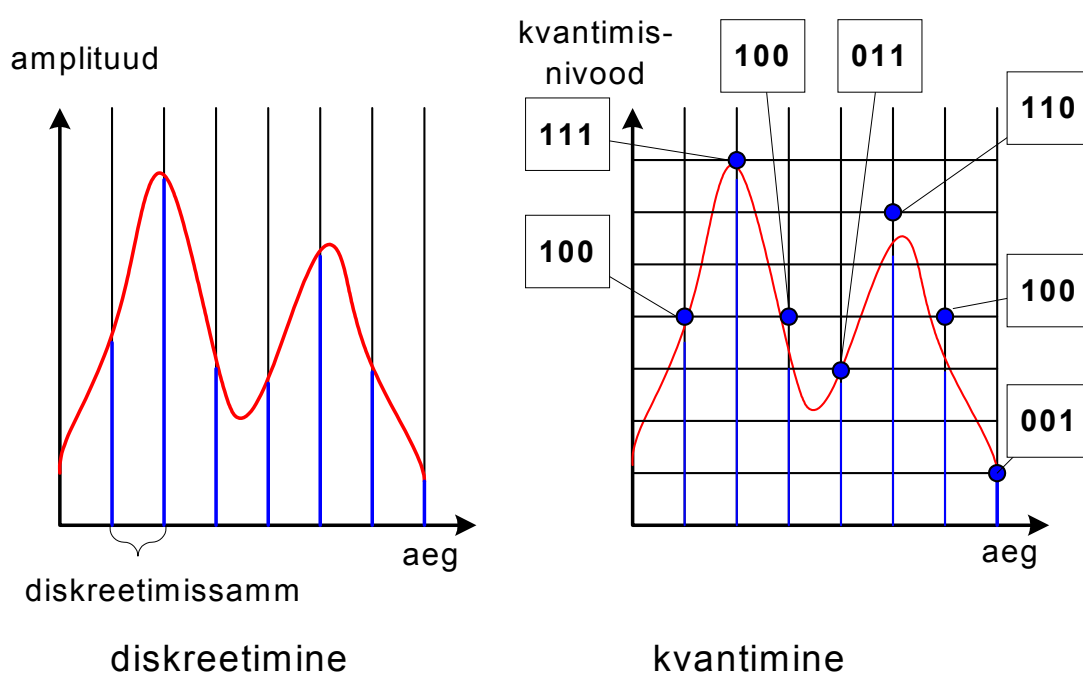
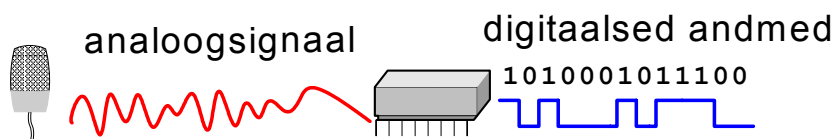
Piiratud sagedusribas oleva analoogkujul helisignaal (inimese kõne) kõigepealt diskreeditakse sagedusega 8 kHz (8000 diskreeti sekundis). Siis kodeeritakse saadud diskreetväärtused erinevatel meetoditel ja kooderitega ning saadakse olenevalt nendest teatud kiirusega andmevoog (5 kb/s kuni 64 kb/s). See andmevoog edastatakse kuulajani, kus see dekodeeritakse ja mängitakse ette. [7]

Heli digitaliseerimine on näitlikult esitatud joonisel (Joonis 1). Diskreedi suuruseks on võetud 3 bitti, mis võimaldab eristada 8 erinevat signaali amplituudi nivood..

Suurema diskreetimissageduse korral on digitaalne tulemus lähedasem algsele helile. Teine parameeter on diskreedi suurus (kvantimisnivoode arv). Mida pikemat andmevälja diskreedi väärtuse salvestamiseks kasutame, seda rohkem erinevaid signaali nivooisid saame eristada ja täpsema vastavuse algsele helile saavutame. 8-bitine diskreet võimaldab salvestada 255 erinevat signaalinivoo väärtust ja 16-bitine 65,535 signaalinivood. [28]

Kodeeritud heli diskreetväärtusi pakitakse, et see võtaks vähem salvestuse mahtu ja edastamisel läbilaskevõimet. Tavaliselt kasutatakse audio pakkimiseks kadudega meetodeid, näiteks ADPCM ja MP3. [28]

Kui kokku on andmeedastuskiirust 128 kb/s või vähem, siis tavaliselt antakse kodeeritud audio jaoks 16 kb/s. Suurematel kogu andmeedastuskiirustel antakse audiolle 64 kb/s, millega saab edastada parema kvaliteediga heli. [7]



Joonis 1. Heli digitaliseerimine

1.2.1 Heli koodekid

Koodek on riistvara või tarkvara, mis teisendab helisignaali digitaalseks koodiks ja vastupidi. Esimesel sammul teisendatakse analoog heli digitaalseteks diskreetideks kasutades impulssmodulatsiooni (PCM) või ADPCM. Järgmisena kasutatakse tajutava

audio kodeerimist digitaalsete andmete hulga edasiseks pakkimiseks. On olemas koodekeid, mis on optimeeritud erinevat liiki helide digiteerimiseks – kõne, muusika jne. Kõnekoodek (*speech codec*, *voice codec* või *vocoder*) on mõeldud inimese hääle jaoks. [28]

Kuna kõne on helidest kõige lihtsam, siis on kõnele optimeeritud koodekitega võimalik saada kõige suuremat pakkimise tihedust ja seega on võimalik selle andmevoogu edastada ka väikese läbilaskevõimega võrgus. Kui aga muusika pakkida kõnekoodekiga, siis on tekkinud kaod liiga suured, et seda muusikat oleks hea kuulata.

Inimkõrv aga ei kuule kõiki sagedusi ühtemoodi, seetõttu on loodud koodekid, mis osa sagedusribast diskreedivad väiksema sammuga kui teist osa.

PCM

Impulssmodulatsioon (*Pulse Code Modulation*, PCM) on videokonverentsi helikoodeki põhistandard. Helisagedusriba, mida diskreeditakse, on telefoniside sagedusriba – 300 Hz...3400 Hz diskreetimissagedusega 8 kHz. Iga väärtus kodeeritakse 8-bitiseks (7-bitiseks, kui edastusvõrgu tingimused piiratud) koodiks. Diskreetimise samm on kogu diskreeditavas sagedusribas ühepikkune. [7]

A- ja μ -Law kooderitega teisendatakse lineaarne kvantimine logaritmiliseks. A-law PCM kooder teisendab lineaarsed 13-bitised PCM diskreedid 8-bitisteks pakitud PCM (logaritmilised) diskreetideks. μ -Law PCM kooder teisendab lineaarsed 14-bitised PCM diskreedid 8-bitisteks [14]

ADPCM

Adaptiivne diferentsiaalne impulssmodulatsioon (*Adaptive Differential PCM*, ADPCM) on arenenum impulssmodulatsiooni tehnika. Iga diskreeidi absoluutse väärtuse kodeerimise asemel kodeeritakse diskreetide väärtuste erinevusest üksteisest. Kodeerimise skaalat saab dünaamiliselt muuta (kompenseerida) vastavalt sageduse või amplituudi muutustele. [28]

CELP

CELP (*Codebook Excited Linear Prediction*) on lainekuju kooder. Korruga diskreeditakse 5 diskreeti ja valitakse grupile koodide hulgast kõige lähedasem. Edastatakse valitud koodi number. Diskreetide võtmiseks kasutatakse A-Law või μ -Law PCM koodekit, diskreedid peab teisendama ühepikkuse diskreetimissammuga diskreetideks. Koode on 1024, seega 10-bitine koodi number 5 diskreediga grupile. Kuna diskreetimise sagedus on 8 kHz ja 5 diskreeti ühendatakse, siis genereeritakse 1600 koodi numbrit sekundis. [7]

LPC

Lineaarse ennustamise meetod (LPC, *Linear Prediction Coding*) kasutab ära seda, et kõne on väga korreleeritud väikses ajaühikus, ja eelmiste diskreetide väärtuste põhjal saab ennustada väikse veaga järgnevad diskreedid.

1.3 Video digitaliseerimine

Video on kujutiste jada, mida näidatakse kiiresti üksteise järel, nii et vaataja tajub terviklikku liikuvat kujutist. Kujutisi nimetame kaadrid ja liikumise kiirust mõõdame kaadrit sekundis.

Kuna üks silmapilgutamine võtab aega 1/50 sekundit e 20 ms, siis inimsilm tajub pidevat liikumist, kui video mängitakse ette kaadrisagedusega 25-30 kaadrit sekundis. Kui kaob mõni kaader, siis vaataja märkab liikumise katkendlikkust. [17]

Meie ülesandes on vajalik videot efektiivsemalt edastada, kui võrgu andmeedastuskiirus pole piisav. Kui mingitest ressurssidest jääb puudu, siis võetakse seda video kvaliteedi arvelt.

Digitaliseerimiseks tuleb kujutis laotada. Laotatud kujutis koosneb punktide ridadest, igal punktil ehk pikselil on heledus, värvus jne. Televisioonikvaliteediga video kaader PAL (*Phase Alternating Line*) koosneb 625 reast, kaadritest edastatakse vaheldumisi paarisread ja paaritud read, kokku 25 täiskaadrit (*interlaced frame*) sekundis (50 poolkaadrit sekundis) [28].

1.3.1 Värvimudelid

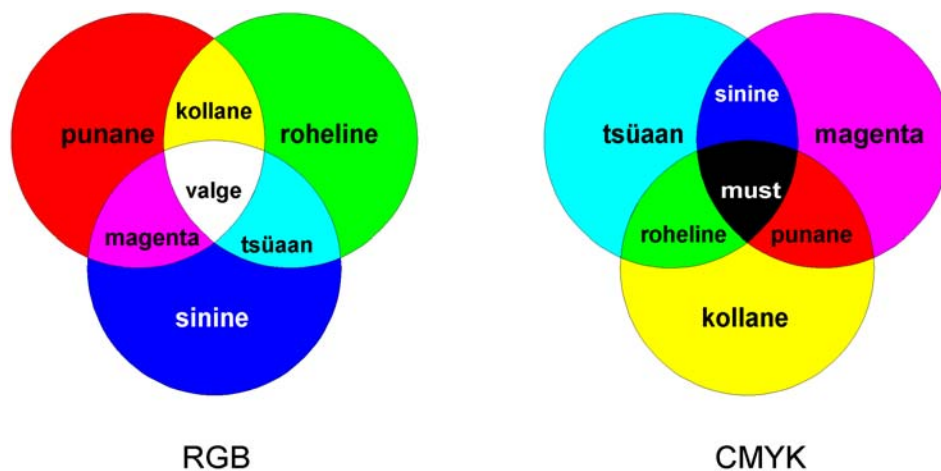
YUV värvimudelit kasutatakse video kodeerimiseks. Y on must-valge signaali heledus. U on punase erinevus (R-Y) ja V on sinise erinevus (B-Y). YUV värvimudeliga moodustatud kujutise kuvamiseks arvutiekraanil peab see olema teisendatud RGB-sse (*red, green, blue*) protsessiga, mis tundub kui värvuste teisendus (*“color space conversion”*). YUV kasutatakse, kuna see säästab salvestuse mahtu ja edastuse läbilaskevõimet võrreldes RGB-ga. YUV kujutis ei ole pakitud RGB, ta on ekvivalentne RGB-ga ja sama andmemahuga. [28]

RGB (*red, green, blue* – punane, roheline, sinine) on värvimudel, mida kasutatakse video ekraanil näitamiseks. Kujutise iga punkti kohta on info punase, rohelise ja sinise värvuse intensiivsuse kohta. Kui kõikide värvuste intensiivsus on suur, saadakse valge. [28]

Värvuse esitamiseks kasutatakse 3 baiti pikseli kohta ja nii saab eristada 16777216 värvuse, seda nimetatakse ka *True color* või 24-bit värvuse esituseks. [28]

CMYK (*Cyan, Magenta, Yellow, black* – tsüaan, magenta, kollane, must) on värvimudel, mida kasutatakse printimisel. Kui kasutatakse värvilisi tinte, siis kõikide värvide suur intensiivsus annab kokku musta värvi. Must värv jääb ilusam, kui selle saamiseks kasutada musta tinti. [28]

Joonisel (Joonis 2) on nende kahe värvimudelite värvide saamine.



Joonis 2. Värvimudelid [28]

1.3.2 Pakkimise meetodid

Vaatleme video pakkimise meetodeid. [1]

1.3.2.1 Ruumiline pakkimine

Ruumilisel pakkimisel pakitakse kokku kaadri visuaalse ala kirjeldus vaadates pikselite mustreid ja kordumisi. Selle pakkimise üks meetod on *run-length encoding* (RLE) (ühesuguste sümbolite jada teisendatakse koodiks [28]) ja seda kasutatakse paljudes koodekites. Suurema pakkimise korral vajalik andmeedastuskiirus ja faili suurus tulevad väiksemad, aga kujutis kaotab teravuse ja värvi infot rohkem. Ruumilise pakkimise tihedust saab kasutaja muuta koodekites kvaliteedi (*quality*) ja vajaliku andmeedastuskiiruse (*data rate*) parameetritega: nende parameetrite väärtuste vähendamine suurendab ruumilise pakkimise tihedust. Mõnedes koodekites on vajalik andmeedastuskiirus ja kvaliteet üksteisega seotud ja ühe muutmine mõjutab teist.

1.3.2.2 Ajaline pakkimine.

Ajaline pakkimine otsib viise, kuidas pakkida kaadrite jadas kaadrite vahelisi muutusi. Analüüsitakse pikselite mustreid ja kordumisi aja jooksul. Iga kaadri iga pikseli kodeerimise asemel kodeeritakse üks kaader täielikult ja iga järgneva kaadri kohta need pikselid, mis erinevad eelmisest kaadrist. Seda tehnikat nimetatakse kaadri eristamine (*frame differencing*).

Kui kaadri enamus pikseleid erinevad eelmise kaadri pikselitest, siis on parem kodeerida terve kaader uuesti. Iga tervelt kodeeritud kaader on nn võtmekaader (*keyframe*), millest alustatakse uut kaadrite vaheliste erinevuste kodeerimist. Paljudes koodekites saab kasutaja määrata võtmekaadri asukoha ajavahemiku järel või määrata kohad, kuhu võtmekaader luua, ja mõned koodekid loovad ise võtmekaadri siis, kui uus kaader on visuaalselt eelmisest väga erinev. Kui on vähem võtmekaadreid, siis andmeedastuskiirus ja faili suurus tulevad väiksemad, aga ka pildi kvaliteet halveneb. Ajalise pakkimise tihedust saab kasutaja tavaliselt määrata koodeki parameetri – video kvaliteedi (*quality*) ja võtmekaadrite arvu valikuga – nende parameetrite väärtuste vähendamisega saavutatakse suurem pakkimine.

1.3.2.3 Kadudega ja kadudeta pakkimine

Kadudeta pakkimisel on kogu algse video info olemas ka pärast lahtipakkimist. Sellega säilitatakse originaali kvaliteet, mis on kasulik video monteerimisel ja transpordil ühest süsteemist teise. Kadudeta pakkimisel võivad faili suurus ja vajalik andmeedastuskiirus tulla liiga suured video ettemängimise süsteemide jaoks.

Kadudega pakkimise meetodid jätavad osa algse video andmeid pakkimise käigus välja. Kadudega pakkivas koodekis saab tavaliselt määrata, kui palju pildi kvaliteeti soovitakse ohverdada väiksema vajaliku andmeedastuskiiruse ja faili suuruse saamiseks. Kadudega pakkimine võimaldab saada palju väiksemaid vajalikke andmeedastuskiirusi ja faili suurusi kui kadudeta pakkimine. Nii on kadudega koodekid kasutusel internetis või andmekandjal asuvatel videotel.

JPEG koodek on alati kadudega ja Planar RGB koodek on kadudeta [1]. Teiste koodekite pakkimise kaod sõltuvad kasutaja seatud vajaliku andmeedastuskiiruse ja kvaliteedi parameetritest.

1.3.2.4 Asümmeetriline ja sümmeetriline pakkimine

Valitud koodek mõjutab ka koodekile vajaminevat aega antud arvu kaadrite pakkimiseks. Kiire pakkimine aitab video loomisel, kiire lahtipakkimine on hea ettemängimisel. Paljud koodekid vajavad palju rohkem aega kaadrite pakkimiseks kui nende lahtipakkimiseks ettemängimiseks. Näiteks 30-sekundiline video võib vajada mitmeminutilist töötlusaega enne ettemängimist.

Koodek on sümmeetriline, kui video pakkimiseks ja lahtipakkimiseks vajatav aeg on ühepikkune. Koodek on asümmeetriline, kui video pakkimiseks ja lahtipakkimiseks vajatavad ajad on erinevad. Näiteks Cinepak asümmeetriline koodek pakib lahti suhteliselt kiiresti, aga vajab pakkimiseks rohkem aega. Koodeki sümmeetrilisus sõltub koodekist ja tavaliselt pole kasutaja poolt muudetav.

1.3.3 Video koodekid ja algoritmid

Video koodek on riistvara või tarkvara, mis teisendab analoogvideosignaali (NTSC, PAL, SECAM) digitaalseks koodiks ja vastupidi.

Mõiste võib viidata ainult pakkimisele ja lahtipakkimisele, mida saab teha tarkvaraga või riistvaraga ja mis on eraldatud analoog-digitaal ja digitaal-analoog teisendusest. [28]

Edasises on lühikirjeldused levinud videokoodekite ja algoritmide kohta.

H.261

H.261 on video tihenduse standard, mis loodi ITU-T poolt ISDN võrgus töötamiseks. Andmed pakitakse 64x kb/s andmeedastuskiirusele, kus x on 1...30, olenevalt kasutatavate ISDN kanalite arvust. See standard loodi peamiselt videokonverentsi ja videotelefoni jaoks. [12]

H.263

H.263 on ITU standard videokonverentsi edastatava andmevoo pakkimiseks. Põhineb H.261 ja on laiendustega, mis parendavad video kvaliteeti üle modemi. H.263 toetab CIF, QCIF, SQCIF, 4CIF and 16CIF resolutsioone. [28]

M-JPEG

M-JPEG (Moving Jpeg) on liikuvate piltide pakkimise meetod, mis rakendab JPEG liikumatu kujutise pakkimise algoritmi igale pildijada kaadrile. Video ettemängimisel vajab arvutit, mis pakiks lahti ja näitaks iga JPEG kujutist parajalt kiiresti, et saavutada sobivat kaadrisagedust (kaadrit sekundis).

M-JPEG jaoks pole standardit, aga on riistvara, mis disainitud töötama televisiooni kaadrisagedusega ja eraldusvõimega. [12]

RLE

RLE (*Run-length encoding*) on pakkimise algoritm, mis asendab järjestikuste korduvate märkide jadad ("run") üksiku märgiga ja jada pikkusega. Seda algoritmi saab rakendada kõikidele sisendmärkidele kaasa arvatud pikkusega 1 või saab kasutada eri märki jadade kodeeritud grupi tähistamiseks. Mida pikemad ja sagedasemad jadad on, seda suurem pakkimine saavutatakse. See tehnika on eriti kasulik musta ja valge kujutise kodeerimisel, kus andmeühikud on pikselid. [12]

Indeo

Indeo on Inteli video pakkimise-lahtipakkimise algoritm. Indeo on Inteli tarkvaraline koodek, DVI on Inteli riistvaraline koodek. [39]

Cinepak

Cinepak on video pakkimise-lahtipakkimise algoritm. Kasutatakse laialdaselt Macintosh arvutites ja on Microsofti Windows 95/98 koosseisus. [28]

Video1

Video1 on Microsofti video pakkimise-lahtipakkimise algoritm [28]

MPEG

MPEG – digitaalse video pakkimise standardite pere. MPEG faile saab dekodeerida riistvaras või tarkvaras. MPEG saavutab suure pakkimistiheduse kodeerides kaadrite vahelisi muutusi kuni järgmise võtmekaadrini. Video info kodeeritakse kasutades DCT (diskreetset koosinusteisendust). MPEG on kadudega pakkimise algoritm, mille tekitatud kaod on tavaliselt inimesele märkamatud.

MPEG standardist on 3 versiooni: MPEG-1, MPEG-2 ja MPEG-4. MPEG-1 saavutab video eraldusvõime 352x240 pikselit ja kaadrisageduse 30 kaadrit/s. See on kvaliteedilt natuke halvem kui tavalise videomaki video. MPEG-2 saavutab eraldusvõimed 720x480 ja 1280x720 kaadrisagedusega 60 kaadrit/s. See on piisav televisioonistandarditele ja kasutatakse DVD-ROM jaoks. MPEG-2 pakib 2-tunnise video 2-4 GB suuruseks failiks. MPEG-2 video pakkimine vajab rohkem arvutusvõimsust kui lahtipakkimine. MPEG-4 on graafika ja video pakkimise algoritmi standard, mis põhineb MPEG-1, MPEG-2 ja Apple QuickTime tehnoloogiatel. Lainekeste (*wavelet*) teisenduse põhise MPEG-4 failid on väiksemad kui JPEG ja QuickTime failid ja nad on mõeldud video edastamiseks kitsamatel kanalitel ning videole saab lisada teksti, graafikat, 2- ja 3- mõõtmelisi animatsioone. [39]

MPEG-1 352x288 @ 30fps koos 24-bit värvi ja CD kvaliteediga heliga ning vajab edastuskiirust 1,5 Mb/s. MPEG-2 720x480, vajab edastuskiirust 4 Mb/s ... 15 Mb/s. MPEG-4 võimaldab palju erinevaid audio ja video edastuskiiruseid. Sisaldab vahendeid intellektuaalse omandi kaitseks. [28]

MPEG-2 töö algoritm.

MPEG-2 kodeerimisel kasutatakse 3 tüüpi kaadreid. I-kaadrid (*intermediate frame*) on nn võtmekaadrid ja sisaldavad terve kujutise. P-kaadrid (*predicted frame*) genereeritakse nendest I-kaadritest või teistest P-kaadritest. Ning B-kaadrid (*bidirectional frame*) interpoleeritakse puhverdatud I- ja/või P-kaadritest. Töötlemise käigus kooder genereerib kõigepealt I-kaadri. P-kaader saadetakse kindla intervalli (tavaliselt 90-100 ms) järel hiljem. B-kaadrid interpoleeritakse kahest eelmisest umbes intervallide 30 ms järel. Niiviisi tekib võimalik kaadrite järjekord: IBBPBBPI. Tavaliselt I-kaadrid genereeritakse iga 400 ms järel ja on eraldatud 10-12 P- ja/või B-kaadritega. [13]

1.4 Multimeedia edastamine

Multimeedia ettemängimiseks vajatakse igal ajahetkel mingi hulk andmeid ja väike varu, mitte aga kogu multimeedia faili. Kuna ka multimeedia failid on tavaliselt mahukad, siis kasutajale on mugavam, et ettemängimine algab varem, kui kogu faili saabumine on lõppenud. Sealjuures peab failist saabuma kindel hulk ajaühiku kohta.

1.4.1 Edastuse transpordi protokollid

Siin vaatleme transpordi protokolle RTCP, RTP, RTSP, TCP ja UDP.

Reaalaja transpordi protokoll – RTP – *Real-time Transport Protocol* – reaalaja andmete transpordiks.

Reaalaja transpordi juhtprotokoll – RTCP – *RTP Control Protocol* – reaalaja andmete transpordi juhtimiseks.

Reaalaja andmevoo protokoll – RTSP – *Real-time Streaming Protocol* – reaalaja andmevoo transpordiks.

Edastusohje protokoll – TCP – *Transmission Control Protocol* – interneti standardne transpordikihi protokoll, ühenduse loomisega ja kinnituste saatmisega töökindel protokoll.

Datagrammiprotokoll – UDP – *User Datagram Protocol* – interneti standardne transpordikihi protokoll, ühendusevaba ja kinnituste saatmiseta protokoll.

1.4.1.1 Reaalaja andmeedastus

Reaalaja andmeedastus on nn ajakriitiline, kus on oluline, et andmed saabuks saatjast vastuvõtjasse õigel ajal. Ettemängimisel ei saa kasutada andmeid, mis saabuvad liiga vara või hilja, puhvris võib nende jaoks ruumi mitte jätkuda, mille tõttu ettemängimise kvaliteet võib halveneda. See oleneb rakendusest.

Multimeedia edastamisel võrgu kaudu andmed saabuvad andmevoona. Faili ettemängimiseks pole vaja kogu faili alla laadida, vaid on võimalik alustada ettemängimist kohe, kui on saabunud soovitud failist vajalik hulk andmeid. Sellist edastust nimetatakse tavaliselt andmevoo (*stream*) edastuseks. Siin on samuti oluline andmete saabumine õiges järjestuses õigel ajal, muidu võivad ettemängimisel tekkida pausid (katkestused).

Sünkroniseerimine ehk ajastamine tähendab saabuva andmevoo ettemängimine selliselt, et ajaliselt ühel ajal loodud andmed saaks ka ette mängitud ühel ajal. Ja kui korraga võetakse mitu loogiliselt kokkukuuluvat andmevoogu näiteks video ja heli, siis ajastamisega saavutatakse sama ajahetke heli ja pildi ettemängimine ühel ja samal ajal.

Reaalaja andmeedastusel on mõned parameetrid, mille muutumine teatud piirides ettemängimise kvaliteeti oluliselt ei mõjuta. Kuna võrgud tavaliselt ei võimalda väga täpselt prognoosida andmete saabumise aega, siis on saabumise lubatud erinevus (*jitter*), mille piiridesse jäämisel samuti ei teki kadu ettemängimise kvaliteedis.

Siin peatükis vaadeldavad protokollid (RTP, RTCP ja RTSP) töötavad OSI mudeli transpordi (4.) kihil. Allasuvatest protokollidest võivad nad töötada IP ning TCP või UDP peal. Võrgus adresseeritakse saatjaid-vastuvõtjaid transpordi aadressidega ehk IP aadressi ja pordi numbri paaridega, tähistab transpordi kihi lõpp-punkti.

1.4.1.2 RTP ja RTCP

RTP (*Real Time Protocol*) pakub lõpp-punktide vahelist reaalaja parameetritega andmete edastuse teenust. Seda tehakse järjestamise ajatemplitega ja edastuse jälgimisega. RTP töötab UDP peal (ühenduse loomiseta) ja kasutab selle kontrollsumma ja multipleksimise teenuseid. RTP paketi päises on andmeploki järjekorranumber, sünkroniseerimise info ja andmepaketi kodeerimise viis. RTP-ga

võib edastada andmeid korraga paljude vastuvõtjateni, kui allasuv võrk võimaldab multisaadet.

RTP protokoll võimaldab erinevat tüüpi andmeid (*payload*) edastada erinevalt. Erinev võib olla heli või video kodeerimine (*encoding*), need reeglistikud on määratud teiste RFC dokumentidega. RTP sessioon on osalejate vaheline assotsiatsioon, mis peab sidet RTP protokolliga.

RTCP (*Real Time Control Protocol*) jälgib RTP teenuse kvaliteeti, annab infot sessioonis osalejate kohta. Selle protokolliga funktsionaalsust võib täita ka eraldi sessiooni juhtimise protokoll. On olemas rakendus, mille tüüp on *monitor* ja mis võtab vastu RTCP pakette RTP sessioonis osalejatelt, hindab vastuvõtjate vastuvõtu kvaliteeti, vigade analüüsi ja statistikat.

RTCP protokolliga saadetakse perioodiliselt juhtinfo pakette sessioonis osalejatele kasutades andmepakettidega sama edastusmehhanismi.

RTCP funktsioonid on:

1. Pakkuda andmeedastuse kvaliteedi kohta tagasisidet. Kasulik adaptiivsel kodeerimisel või aitab võrguteenuse pakkujal võrgu probleeme diagnoosida.
2. RTP allika identifikaatorit nim *Cname* (kanooniline nimi). Vastuvõtjad vajavad *Cname* osalejate jälgimiseks ja kui sessioonis on mitu andmevoogu, siis nende osalejatega seostamiseks, näiteks audio ja video sünkroniseerimiseks.
3. Iga osaleja peaks saatma RTCP pakette kõikidele osalejatele, iga osaleja saaks sõltumatult jälgida teisi osalejaid.
4. Vähi vajaliku sessiooni juhtimise info edasi andmine, näiteks osaleja identifikaatori näitamine kasutajaliidesel.

Funktsioonid 1-3 on kohustuslikud RTP kasutamisel.

RTCP paketi tüüpideks on saatja raport, vastuvõtu raport, allika (saatja) kirjeldus, sessioonist lahkumine ja rakenduse määratud tüüp. RTCP pakettis on sessiooni kirjeldus, mis sisaldab sessiooni nime, kasutaja nime, kasutaja e-posti aadressi, telefoni numbrit, geograafilist asukohta, tarkvara nime jm. Seda infot võib rakenduse kasutajaliides sessiooni kasutajate kohta kasutajale näidata.

RTP turvalisusest. Ainult volitatud kasutajad saaks osaleda RTP sessioonis – autentimine ja terviklikkus – loodetakse teostada madalama taseme protokollidega (transpordi ja võrgu kihtidel) tulevikus. Konfidentsiaalsus saavutatakse andmete krüptimisega.

RTP ei reserveeri ressursse ega garanteeri teenuse kvaliteeti. RTCP juhib andmeedastust. Mõlemad protokollid on sõltumatud allasuvatest transpordi ja võrgu kihtidest. [25]

1.4.1.3 RTSP

RTSP (*Real Time Streaming Protocol*) on rakendustaseme protokoll reaalaaja omadustega andmete edastuse juhtimiseks. Töötab RTP, UDP, multisaate UDP ja TCP peal.

Kliendi ja serveri vahel vahetatavate käskudega on võimalik luua sessioon, kokku leppida sessiooni parameetrid, ajutiselt andmevoo saatmist katkestada sessiooni

katkestamata, sessiooni andmeid salvestada, sessiooni käigus klienti teise serveri juurde ümber suunata ja sessioon lõpetada. Andmevoo edastamise küsimustega tegelevad alumised protokollid: RTP, transpordi ja võrgukihi protokollid, teenuse kvaliteedi tagamise protokollid jne.

RTSP on laiendatav, uusi parameetreid ja meetodeid saab lisada. Transpordi meetod kliendi ja serveri vahel kokku lepitud (näiteks andmeedastuse kiirus, transpordi protokoll jne). RTSP kasutab veebi (HTTP) turvamehhanisme ja on analüüsiv HTTP ja MIME (*Multipurpose Internet Mail Extensions*) analüsaatoritega. Andmete (*payload*) tüübid defineeritud MIME tüüpidega. RTSP on multiserver võimalusega st meediavood võivad asuda erinevatel serveritel. Klient loob ühenduse kõikide vajalike serveritega. Sünkroniseerimine tehakse transpordi kihil.

Kuna RTSP kasutab HTTP (veebiserveri) turvamehhanisme, siis on ohud sarnased. Sessioonis osalejaid soovatakse autentida ja kahtlase päringu korral soovatakse vastata HTTP veateatega “403 Forbidden” (juurdepääs keelatud).[26]

1.4.1.4 TCP ja UDP

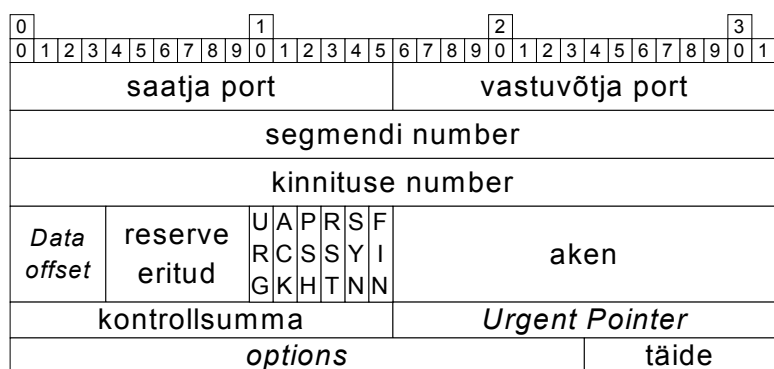
TCP (*Transmission Control Protocol*) [21] on ühenduse loomisega lõppkasutajate vaheline töökindel transpordi protokoll. TCP pakub töökindlat protsesside vahelist sidet arvutites, mis on ühendatud arvutivõrkudesse. TCP eeldab, et saab alumiste kihtide protokollidelt lihtsa, potentsiaalselt mittetöökindla datagrammi teenuse. TCP tegeleb ühenduste loomisega ja lõpetamisega ning ühenduste ajal andmete edastamisega.

TCP pakub järgmisi teenuseid:

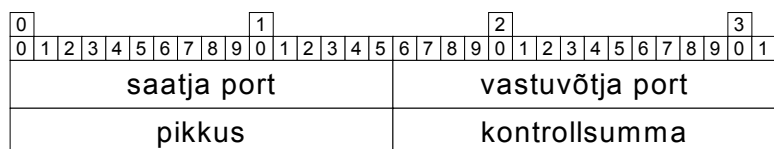
- andmeedastus – TCP on võimeline edastama pidevat baitide voogu igas suunas oma kasutajate vahel pakkides baidid mitmekaupade segmentidesse. TCP otsustab, millal andmeid edastada ja millal oodata.
- töökindlus – TCP peab taastama andmed, mis on riknenud, kadunud, topelt või pole õiges järjekorras. Selleks antakse igale edastatavale baidile järjenumbr ja nõutakse sellele vastuvõtjalt kättesaamise kohta positiivset kinnitust (ACK). Kui kinnitus ei saabu teatud ajaintervalli jooksul, siis andmed edastatakse uuesti. Vastuvõtjas järjestatakse saabunud segmendid kasutades järjenumbreid. Riknemine avastatakse iga edastatud segmendi kontrollsummat vastuvõtjas kontrollides ja riknenud segmendid eemaldatakse.
- voo juhtimine – saatja saab vastuvõtja andmete saatmise hulka juhtida seades “akna” kus iga kinnitus näitab järgmist segmendi numbrit, milleni võib saatja saata.
- multipleksimine – et ühes hostis saaks palju protsesse kasutada sidet samaaegselt, siis antakse protsessidele aadressid ehk pordid. Pordi number ja IP aadress koos moodustavad transpordi aadressi (*socket*). Transpordi aadresside paar identifitseerib iga ühenduse. Iga host käsitleb portide ja protsesside sidumist ise, tuntud teenustele antakse igas hostis sama pordi number.
- ühendused – iga ühendus on unikaalselt identifitseeritud aadressi ja pordi paariga. TCP haldab iga andmevoo (protsessi) olekut. Ühenduse moodustavad andmevoo olek, transpordi aadressid, järjenumbrid ja akna suurused. Kui 2 protsessi soovivad omavahelist sidet, siis TCP peab looma ühenduse, ja kui protsessid sidet enam ei vaja, siis TCP peab ühenduse lõpetama ja ressursid vabastama.

Igale baidile antakse järjenumber. Iga segmendi järjenumber on esimese andmebaidi järjenumber. Kinnituse number on järgmise oodatava vastassuunas edastatava segmendi järjenumber. Kui TCP edastab andmesegmendi, siis ta paneb segmendi koopia uuesti edastamise järjekorda ja ootab. Kui selle segmendi kohta saabub kinnitus, siis segment kustutatakse järjekorrast; kui kinnitust ei saabu enne taimeri lõppu, siis see segment edastatakse uuesti. Kinnitus ei garanteeri, et andmed on edastatud lõppkasutajale, vaid seda, et vastuvõtja TCP on saanud andmed kätte.

UDP (*User Datagram Protocol*) on mõeldud datagrammi režiimiks arvutite vahelises pakettisides. Eeldab, et allpool asub IP protokoll. UDP võimaldab rakendusel saata sõnumeid teistele programmidele minimaalse protokollimehhanismiga. Protokoll on tehingule orienteeritud, edastamise ja koopiade tekkimise kaitse pole garanteeritud. Pakettide kättesaamise kinnitusi ei saadeta. UDP moodul määrab saatja ja vastuvõtja IP aadressid ja protokollivälja. [22]



TCP paketi päis



UDP paketi päis

Joonis 3. TCP paketi päis ja UDP paketi päis.

Kui võrrelda TCP ja UDP päiseid (Joonis 3), siis mõlema protokollipäises sisalduvad portide numbrid ja kontrollsumma. TCP päis on vähemalt 24 baidi pikk ja UDP päise pikkus on 8 baidi. TCP päises sisalduvad saatetava ja kinnitatava segmendi järjenumbrid, akna suurus ja ühenduse hoidmisega seotud bitid. UDP päises on veel ainult paketi pikkus.

Rohkem infot interneti protokollide kohta on [27].

1.4.2 Multisaade

IP multisaade võimaldab otstarbekamalt kasutada võrgu läbilaskevõimet. Multimeedia andmevoo vastuvõtjaid adresseeritakse ühe ja sama grupiaadressiga (IP klass D), millega välhitakse igale vastuvõtjale eraldi andmete koopia saatmist.

IP multisaade on IP datagrammi edastamine hostide grupile, mis on identifitseeritud ühe aadressiga. Hostide grupp on dünaamiline, grupi suurusele pole piirangut. [6]

Kasutatakse D-klassi IP aadresse vahemikust 224.0.0.0 – 239.255.255.255 [15]

Kasutatakse protokolliga IGMPv2 (*Internet Group Management Protocol*) [10]. Multisaate marsruuterid peavad oma võrkudes asuvate hostide kohta nimekirju, milline host millises grupis on ja hoiavad seda nimekirja ajakohasena küsides perioodiliselt hostilt, kas ta on grupis. IGMP protokoll kasutab multisaate aadressi 224.0.0.1 üldisteks päringuteks ja aadressi 224.0.0.2 kasutavad hostid grupist lahkumisest teatamiseks kõikidele marsruuteritele.

See marsruuter, mille ülesanne on hoida gruppe, pärib grupi liikmeid iga 125 sekundi järel ja ootab vastust 10 sekundi jooksul. 260 sekundit peab mööduma, enne kui marsruuter saab aru, et grupis pole liikmeid.

Multisaade pole üle interneti lubatud. Arvatav põhjus võib olla selles, et multisaate kasutamine on efektiivne väga suurte andmehulkade edastamisel paljudele vastuvõtjatele ja multimeediast on võimalik luua väga suuremahulisi andmevooge, mis võivad interneti ummistada.

On olemas nn multisaate “saarekesed” – kohtvõrgud, mille sees saab multisaadet kasutada, aga nende vahel ei saa. Kui mitu sellist saarekest tahavad koos multisaadet kasutada, vajatakse multisaate tunnelit nende võrkude vahele.

Multisaate tunneli protokoll on väljatöötamisel ja selle nimi on *UDP Multicast Tunneling Protocol* (UMTP). UMTP saab olema mõeldud MBone või teiste multisaate võrkudele juurdepääsuks neile hostidele, mis neid interneti ühendavate marsruuterite pärast ei saaks multisaate võrku kasutada. Kasutades UDP protokolliga saab teha rakendusetaseme programmi, mis tunneldamist teostavad ilma hosti operatsioonisüsteemi muutmata. [11].

1.4.3 Videokonverents

Videokonverentsi standard H.323 koosneb soovitud seadmetest videokonverentsis kasutamiseks, kui allasuv võrk ei paku garanteeritud teenuse kvaliteeti.

H.323 terminal koosneb[20]:

1. Helikoodekid:
 1. G.711 – PCM audio koodek edastuskiirusega 56/64 kb/s,
 2. G.722 – ADPCM audio koodek 7 kHz edastuskiirusega 48/56/64 kb/s,
 3. G.723 – kõnekoodek edastuskiirusega 5,3 ja 6,4 kb/s,
 4. G.728 – kõnekoodek edastuskiirusega 16 kb/s,
 5. G.729 – kõnekoodek edastuskiirusega 8 või 13 kb/s.
2. Videokoodekid:
 1. H.261 – andmeedastuskiirusel ≥ 64 kb/s ja
 2. H.263 – andmeedastuskiirusel < 64 kb/s.
3. süsteemi juhtimine:
 1. H.245 – süsteemi juhtimise protokoll,
 2. H.235 – turvalisus,

3. Q.931 – abonentide signaaliseerimine (*digital subscriber signalling*),
 4. H.225.0 – kõne juhtimise protokoll (*call control protocol*),
 5. H.450.1 – H.323 lisateenuste toetuse protokoll.
 6. RAS (*Registration, Admission and Status*) juhtimine H.225.0
4. andmekonverentsi standard T.120 (faili edastus, liikumatu kujutise edastus, valge tahvli jagamine).

Videokonverentsis kasutatavad kaadri suurused ja edastuskiirused

CIF (*Common Intermediate Format*) – video kaadri formaadi standard, mida kasutatakse videokonverentsis.

CIF formaat	Kaadri suurus	Edastuskiirus kaadrisagedusel 30 kaadrit/s (Mb/s)
SQCIF (<i>Sub Quarter CIF</i>)	128 x 96	4.4
QCIF (<i>Quarter CIF</i>)	176 x 144	9.1
CIF (<i>Full CIF, FCIF</i>)	352 x 288	36.5
4CIF (4 x CIF)	704 x 576	146.0
16CIF (16 x CIF)	1408 x 1152	583.9

Tabel 1. Kaadri suurused ja edastuskiirused [28]

1.4.4 Multimeedia andmevoogude eriomadused ja nõuded võrgule

Multimeedia andmevoog vajab võrgult garanteeritud andmeedastuskiirust ja eeldab viite muutust teatud piirides. Kui võrgutingimused muutuvad, siis multimeedia rakendused on võimelised käimasoleva sessiooni käigus neid parameetreid ka muutma. Kasutaja näeb ja kuuleb siis ettemängitava video ja audio kvaliteedis muutust. Inimene tajub väikseid katkestusi – vaikusehetki helis ja seisvat pilti videos. Seepärast ei tohi andmete saabumise viite muutus teatud piire ületada.

1.4.4.1 Sisu nõuded

Sisu nõuded olenevad sellest, mida vastuvõetava multimeediaga tehakse, näiteks kui suurel ekraanil seda vaadatakse. Siin vaatleme püstitatud ülesandest tulevaid nõudeid.

Vastuvõtjaid on 2 tüüpi – heade võrgutingimustega ja halbade võrgutingimustega või heade ettemängimise võimalustega ja halbade ettemängimise võimalustega (näiteks väike või must-valge ekraan). Mõlematele peab leidma sobivama lahenduse. Edastusvõrgult nõuame muutumatut viidet ja garanteeritud andmeedastuskiirust. Kui on vaja millegi pealt selles multimeedias kokku hoida, siis teeme seda videopildi arvelt.

Heli nõudeks on parim võimalik. Heli sisu on peamiselt inimese kõne, mille jaoks loodud kodekid suudavad heli digitaliseerida üsna väikese edastuskiiruse jaoks.

Loengu olulisem osa on kuulamine. Materjalid antakse interneti kodulehel allalaadimiseks.

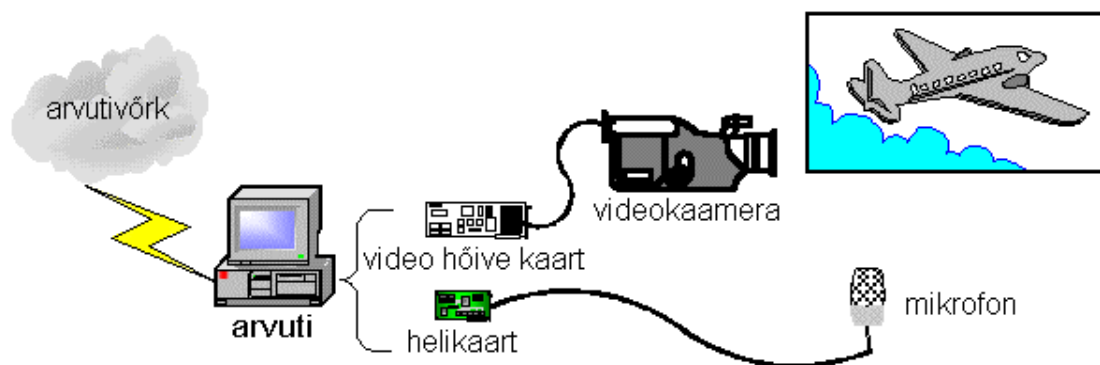
Kui vaadatakse suuremal ekraanil, siis pilt peab olema kvaliteetne, kui väiksel ekraanil, siis võib edastatav kaadri suurus olla väiksem ja sellest saadakse suur kokkuvõtte edastatavas andmehulgas.

Õppejõud on oma loengus üsnagi liikuv ja mõnikord on vaja näha näiteks ta käeliigutusi või kui ta midagi tahvlile kirjutab. Loengu käigus tekkinud materjalid on hiljem olnud kättesaadavad küll ka interneti kodulehelt pildifailidena.

Vastavalt vastuvõtja tüübile on nõutav video kvaliteet. Edastatava kujutise suurus: kui loenguruumis seinale näidata, siis suurim võimalik; kui arvutiekraanilt vaadata, siis CIF (352 x 288) sobib ka. Kaadrisagedus arvutiekraanil – võib ka 8 kaadrit/s rahule jääda.

1.5 Multimeedia riistvara

Multimeedia andmete loomise (digitaliseerimise) arvutis vajatakse heli ja videohõive (*capture*) ja digitaliseerimise riistvara (Joonis 4). Audio jaoks helikaart, millel on sisend mikrofonile või liinisissend (*line in*). Video jaoks on vaja kaamerat ja liidest kaamera arvutiga ühendamiseks. Toodetakse ka selliseid veebikaameraid, mis ühendatakse arvuti USB porti, nende jaoks pole eraldi riistvaralist liidest vaja.



Joonis 4. Arvutile lisatav vajalik riistvara heli ja video jaoks

Siin alapeatükkides on mõned kogemused kasutatud riistvara kohta.

1.5.1 Audioseadmed

Heli seadmetest kasutati arvutitesse installeeritud seadmeid. Nende kasutamine oli lihtne ja mugav ning probleemideta. Heli salvestamisel salvestamise tundlikkus ei peaks olema maksimaalne, siis võib helikvaliteet halb tulla.

Tihti on videohõive kaartidel helisisend ka. Kui kasutada sama riistvara nii heli kui video jaoks, saadakse parem sünkroonsus heli ja video vahel.

Kui arvutis on mitu heliga tegelevat riistvara, siis peab tarkvaras määrama, millist kasutatakse või kontrollima tarkvarast riistvara valiku seadistusi. Helikaardi sisendi (mikrofoni või liini sisendi) valikul peab vaatama, millisesse sisendisse on kaabel ühendatud ja tarkvaras tuleb ka see sisend arhiveerida.

1.5.2 Videoseadmed

Kogemused on olemas järgmise riistvaraga:

- Connectix quickcam. Max pildi suurus 320x240 px, mustvalge (64 halltooni), 30 kaadrit/s. Keskmise kaadri maht 76800 B.

- Creative webcam. USB porti. Max pildi suurus 640x480 px, 15 kaadrit/s, värviline RGB. Keskmine kaadri maht 921600 B.
- AV Master. Max pildi suurus 768x576 px, 2 välja. Kaadri pakkimistihedus 4:1 (andmevoog 5,5 Mb/s)...100:1. MPEG või AVI: 1:27,0 (8 kb/väli, 0,2 Mb/s, 11 Mb/min) ... 1:3,9 (56 kb/väli, 1,4 Mb/s, 82 Mb/min).
- Video Capture Card BT7x7 – selle kohta pole palju teada.
- Winnov Videum AV. Max pildi suurus 4CIF (704x576), 25 kaadrit/s, YVU9: YUV, 4:2:0, Intel Indeo Raw.

Kõige lihtsam oli kasutada Creative webcam. See kaamera töötas Microsoft Netmeetingu, vic'i ja RealProduceriga. Tarkvara oli mugav kasutada ja töötas vaikeseadetega.

AV Master ei töötanud RealProduceriga, sest polnud sobivat videotüüpi.

Video Capture Card oli nn odav kaart Brooktree kiibiga. Selle juures oli tähelepanek, et kui arvuti ei olnud töötanud ja siis oli vaja RealProduceriga umbes 1,5-tunnine multimeedia klipp teha, siis selle tegemise ajal tekkis arvuti krahh. Pärast arvuti *restarti* töötas lõpuni korralikult. Kui aga enne multimeedia klipi tegemist oli arvuti töötanud natuke aega (umbes vähemalt pool tundi), siis sai 1,5-tunnise klipi ühekorraga teha. Põhjus võib olla kasutatud riistvara (videohõive kaardi) töö katkemises võibolla liiga kiire temperatuuri tõusu tõttu ja sellise vea (riistvara töö katkemise) mittekäsitlemisest videohõive kaardi juhtimise tarkvara või arvuti operatsioonisüsteemi poolt.

Winnov Videum kaarti sai kasutatud operatsioonisüsteemiga Windows ME (teisi kasutati Windows 98-ga), mis töötas arvutil protsessoriga Pentium4 taksagedusel 1,4 GHz. Protsessori ülekoormuse tõttu tekkisid kaod ja vähenes kaadrisagedus. Kui pilt oli väheliikuv, siis oli protsessor koormatud ja kaadrisagedus suurem. Kui aga pildis oli palju liikumist, siis oli kaadrisagedus väiksem ja ka protsessori koormus väiksem. Oleks oodanud, et väheliikuva pildi korral on protsessori koormus väiksem, kuna kaadritevahelised erinevused on väikesed (kui kasutatakse diferentsiaalset pakkimist). Winnov Videum kaardil on ainult üks RealProducerile sobiv värvimudel – YVU9.

1.6 Tarkvara

Multimeedia andmevooga rakendused võib jagada 2 liiki: need, mis edastavad reaalaajas, et vastuvõtjas saab ette mängida andmeid (peaaegu) samal ajal, kui neid andmeid saatjas luuakse. Teine liik on nn simuleeritud reaalaeg, kus andmed mängitakse vastuvõtjas ette samas järjestuses ja suhtelises ajastuses, kui nad saatjas kunagi loodi. Multimeedia edastamise ja ettemängimise rakendused on videokonverents ja video nõudmisel. Videokonverents on interaktiivne, kui 2 või rohkem osalejat võivad saata oma heli ja videot teistele.

Töös vaatleme 3 tarkvara: Microsoft Netmeeting, Mbone Tools ja Real Networks.

1.6.1 Microsoft Netmeeting

See tarkvara, võib öelda, on kasutajale kõige kättesaadavam, sest maailmas kõige kasutatavam operatsioonisüsteem on ikkagi Microsoft Windows, mille koosseisus see tarkvara on. Tarkvara on tehtud väga lihtsalt kasutatavaks ja optimeeritud väikse andmeedastuskiirusega võrkudele. Heli parameetritest saab valida täisdupleks või mitte, st kas ise saaks samal ajal rääkida, kui kuulan teist. Ülejäänud parameetritest

saab valida, kas teha automaatselt või käsitsi: häälestada heli tugevust, vaikuse taset (kui on vaikus, siis mitte andmeid saata), heli koodekit arvutisse installeeritute seast. Video seaded on samuti lihtsad: valida, kas videot saata ja/või vastu võtta. Saatmisel kujutise suurus (väike, keskmine, suur) ja video seade arvutile installeeritute seast. Vastuvõtmisel valida, kas soovitakse kiiremini liikuvat videot (suurem kaadrisagedus) või kvaliteetsemat kujutist. Vastavalt sellele vastuvõtja parameetritele seatakse saatja pool saadetakse andmeedastuskiirus.

Kasutaja kohta (iga kasutaja enda kohta) on võimalik esitada eesnimi, perenimi, e-posti aadress, geograafiline asukoht ja kommentaar. Oma võrgu kiiruse parameetritest on valida 4 vahel: 14,4 kb/s modem, 28,8 kb/s modem, xDSL või ISDN ja kohtvõrk. Olemas tarkvara modemiga teistele helistamiseks ja ka arvutivõrgu kaudu "helistamiseks". Arvuteid adresseeritakse NetBIOS nime või IP aadressi järgi.

Netmeeting kasutab heli ja videokonverentsi standardit H.323, andmekonverentsi standardit T.120. Kaamera või videohõive (*capture*) kaart peab toetama standardit *Video for Windows*. [18]

Programme saab jagada kõikide kasutajatega ja jagatud programmi juhtimist anda üle teisele kasutajale (mis on kasutaja hooletuse korral turvaauk tema arvutis).

Ühiskasutusega valge tahvel (*Whiteboard*) – üks kasutaja algatab, igaüks saab valgele tahvlile joonistada ja kõik osalejad näevad.

Chat – jututuba, üks algatab, kõik näevad ja saavad tippida sõnumeid, mida kõik osalejad lugeda saavad.

Videot saab saata 1 saatja 1 vastuvõtjale korraga. Kasutatakse video koodekit Indeo (või sobiv YUV või I420, I420 (sama mis IYUV) on *pixel format* 12 bitti pikseli kohta [8]).

1.6.1.1 Kokkuvõtteks.

Netmeeting on hea oma lihtsa kasutatavuse pärast, aga töö ülesande täitmiseks ei sobi, sest reaalsaja videot saab vahetada ainult 1 kasutajaga; sessiooni videot salvestada ei saa (vaatajate jaoks loeme mitte oluliseks puuduseks), salvestatud videot selle tarkvaraga reaalsajas ette mängida ei saa.

On olemas Microsofti süsteem veebibrauserist vaadata reaalsajas toimuvaid sündmusi, ja neid ka salvestatult vaadata veebilehelt, aga see töötab vaid Microsofti veebibrauseriga.

1.6.2 Mbone Tools

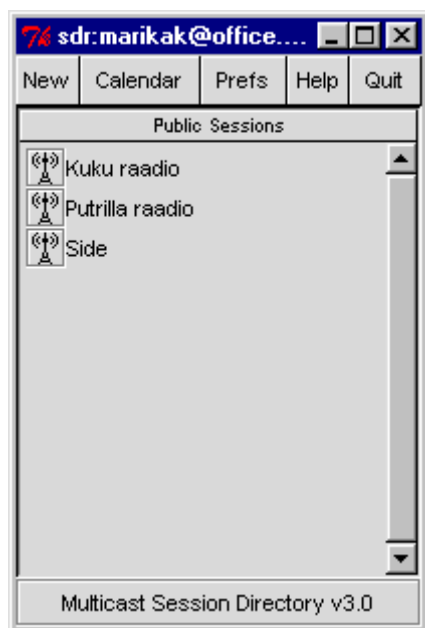
Tasuta kasutada tarkvarade pere, mis kasutab RTP ja RTCP protokolle UDP (pordi number suurem 5000-st) peal ning multisaadet. Koosneb eraldi programmidest sessioonide haldamiseks (sdr), heli (rat) ja video (vic) saatmiseks-vastuvõtuks, valge tahvli jagamiseks (wbd) jm.

Tarkvara on porditud paljudele operatsioonisüsteemidele. Kasutajale antakse sessiooni ja andmete kohta väga palju infot, seetõttu tarkvara on hea näiteks selle ala tundmaõppimiseks. Mbone tarkvara koosseisu kuuluvad programmid moodustavad täiesti omaette keskkonna, mis kasutab multisaadet, kuid tavaliselt multisaade on Interneti võrkude vahel blokeeritud (liiga suure multimeedia poolt tekitatava koormuse tõttu), mis piirab kasutajate ringi palju.

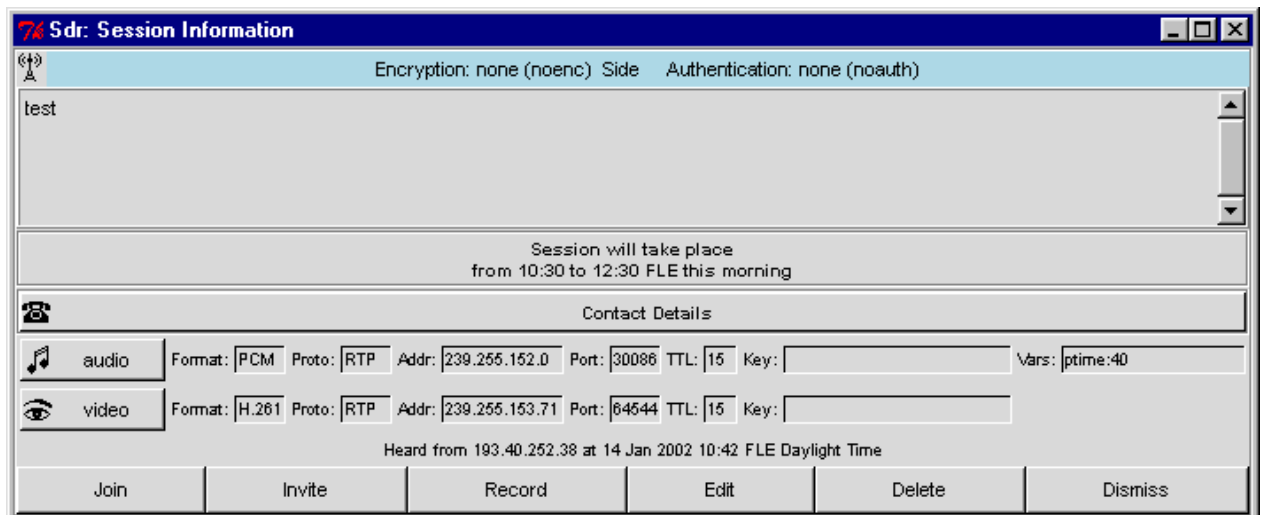
Tarkvara on loodud Berkeley laboratooriumi arvutivõrgu uurimisgrupis (*Network Research Group at the Lawrence Berkeley National Laboratory*) koostöös Kalifornia ülikooliga (University of California, Berkeley). Tarkvara asub <http://www-mice.cs.ucl.ac.uk/multimedia/software/>.

1.6.2.1 sdr

Session Directory (sdr) haldab sessioone. Kasutajad saavad oma sessioone välja kuulutada ja väljakuulutatud sessioonides osaleda (*join*) ja osalejaid kutsuda. Vastavalt sessiooni parameetritele avatakse sessioonis osalejale automaatselt vajalikud Mbone Tools programmid. Kuna multisaate levik on piiratud, siis sessioonide arv oleneb sama võrgu kasutajate aktiivsusest sessioonide loomisel.



Joonis 5. SDR aken sessioonide nimekirjaga. Võrgus töötab 3 sessiooni, nimedega Kuku raadio, Putrilla raadio ja Side, need on mõeldud kõigile (broadcast).



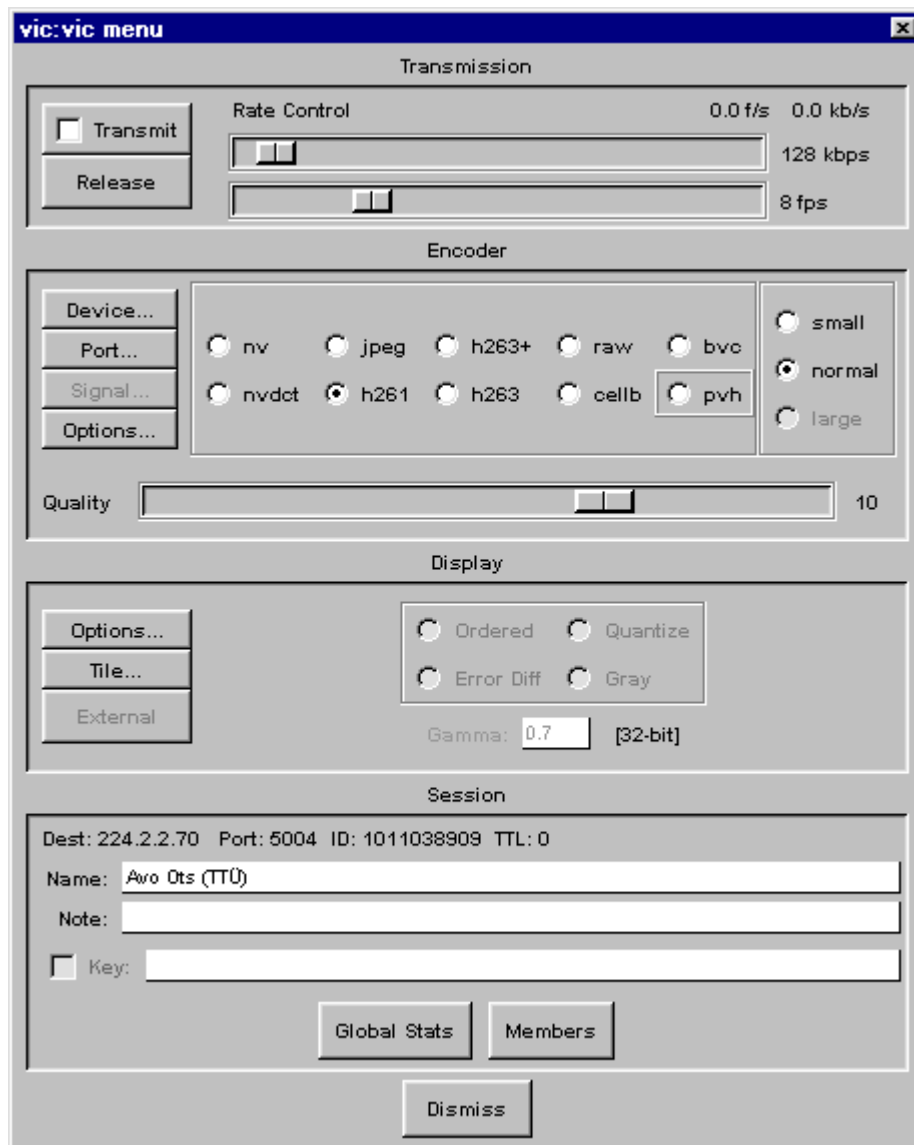
Joonis 6. Ühe sessiooni info: toimumise aeg 14. jaanuar 2002 kell 10.30-12.30, sessioonis on 1 audio andmevoog formaadis PCM, edastatakse multisaate aadressil 239.255.152.0 ja port 30086, TTL 15, krüpteerimata, ptime (packet time) 40 ms; 1 videovoog, formaat H.261, edastatakse multisaate aadressil 239.255.153.71 ja port 64544, TTL 15, krüpteerimata. Sessiooni saadab arvuti IP aadressiga 193.40.252.38. (Tegelikult on see selle arvuti IP aadress, kust see sessioon loodi.)

Uue sessiooni loomisel antakse sessiooniga seotud andmevoogudele (audio, video jm) oma multisaate aadressid ja pordi numbrid (UDP ja suurem kui 5000). Kui ühes hostis on sdr rakendusprogrammiga sessioon defineeritud, siis teises hostis sdr programmi aknas on see sessioon varsti näha. Sessiooni parameetritest saab näha sessiooni tüüpi (*test*, interaktiivne – *meeting*, levisaade, mitteinteraktiivne – *broadcast*, *directory*) – sellele vastavalt seatakse TTL, nime, käitamise aega (algus, lõpp ja kui on perioodiline), saatja hosti (kust see sessioon loodi), meedia andmevoogude tüübid, multisaate aadressid ja pordid, sessiooni kirjeldus ja URL jm. Loodud sessiooni kuulutatakse nii kaua, kuni ta aegub või kustutatakse samast hostist, kust sessioon loodi. On võimalik ka osalejate autentimisega ja krüpteeritud sessioone luua. [29]

1.6.2.2 vic

Video Conferencing tool (vic) on videopildi saatmise, vastuvõtu ja ettemängimisega tegelev programm. Videopildi saatmise parameetreid saab valida väga palju.

Programmi käivitamisel peab käsureale ette andma vähemalt multisaate (või teise hosti) IP aadressi ja pordi numbri (UDP, suurem kui 5002), mis võib olla ebamugav. Kõikides hostides samade käsurea parameetritega käivitatud vic programmid “näevad” üksteist ja võivad saata ja vastu võtta videopilti. Lisaks näevad nad teiste kasutajate nime, hosti nime, saadetava videopildi parameetreid ja videopilti väikses aknas, mida saab suurendada (QCIF, CIF või SCIF, 1/16, 1/4 ja 1 PAL ja NTSC). Tihti kasutatud parameeter oli TTL, millega saab paketi levikaugust määrata. Selleks käsureale lisada -t ja sobiv TTL number (mitu marsruuterit pakett võib oma teel läbida saatjast vastuvõtjani). Kasutati seda selleks, et väga suur andmevoog (1 Mb/s) 2 auditoriumi vahel ei koormaks ülejäänud arvutivõrke. [31]



Joonis 7. Vic aken saadetava videopildi parameetrite määramiseks.

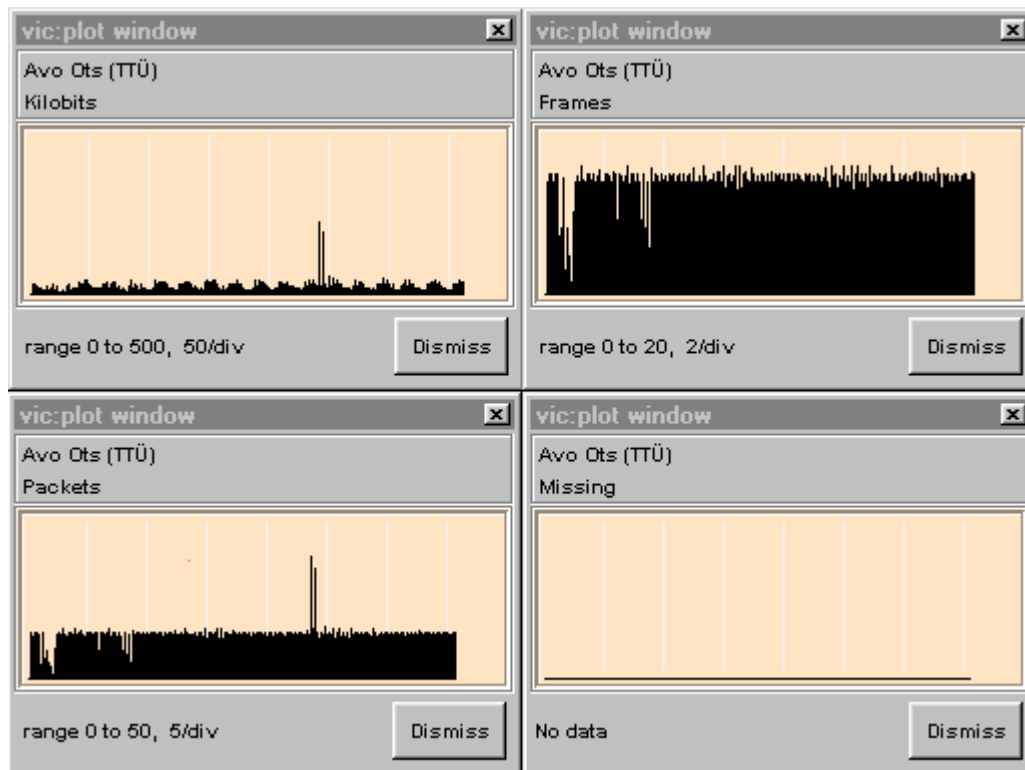
Video saatmiseks peab olema arvutile installeeritud videokaamera ja videohõive kaart (*frame grabber*). Video vastuvõtuks pole vaja erilist riistvara.

Vic näitab statistikat arvuliselt ja graafikutena; andmeid kogub iga sekund.

vic:Avo Ots (TTÜ)			
Avo Ots (TTÜ)			
RTP Statistics			
	EWA	Delta	Total
Kilobits	37.9	47.3	15069
Frames	15.0	14.5	5822
Packets	15.0	14.5	5945
Missing	0.0	0.0	1
Misordered	0.0	0.0	0
Runts	0.0	0.0	0
Dups	0.0	0.0	0
Bad-S-Len	0.0	0.0	0
Bad-S-Ver	0.0	0.0	0
Bad-S-Opt	0.0	0.0	0
Bad-Sdes	0.0	0.0	0
Bad-Bye	0.0	0.0	0
Playout			0ms

Dismiss

Joonis 8. Joonis. Vic statistika. Andmeedastuskiirus (Kb/s), kaadrisagedus (kaadrit sekundis), paketedastuskiirus (paketti sekundis), kadunud paketid. Keskmine, viimane, summaarne.



Joonis 9. Joonis. Statistika ajalised graafikud.

1.6.2.3 rat

Robust Audio Tool (rat) on heli saatmise, vastuvõtu ja ettemängimisega tegelev programm. Heli saatmisel saab palju parameetreid valida ja jälgida sessioonis osalejate vastuvõtu kvaliteeti graafiliselt. Selle programmi käivitamisel peab samuti käsureal ette andma vähemalt multisaate (või teise hosti) IP aadressi ja pordi numbri, mis võib olla ebamugav. Teiste kasutajate kohta näeb nende nime ja hosti nime ning saatjate ja vastuvõtjate kvaliteeti.

Rat kasutamiseks on vaja arvutile internetiühendust ja helikaarti. Audio edastamiseks mikrofoni ja vastuvõtuks (kuulamiseks) kõlareid või kõrvaklappe. Samaaegselt saatmiseks ja vastuvõtuks on parem kasutada kõrvaklappe, sest kõlarid võivad segada mikrofoni. Kasutab IP multisaadet, RTP UDP/IP peal transpordi protokolliks ja vastab RFC1890 (RTP profiil minimaalse juhtimisega audio ja video konverentsi jaoks). Rat kasutab erineva andmeedastuskiiruse ja kvaliteediga koodekeid, vastuvõtja juures paketikadude peitmist ja saatja poolset kanali kodeerimist liiasusega audio edastamiseks. Liiasusega edastamine toimub nii, et igasse paketti pannakse eelmise paketi kokkupakitud koopia. Paketikao korral kannatab heli kvaliteet, aga on siiski parem kui ei oleks kokkupakitud paketi koopiat. Teine viis paketikadu peita on andmevoog plokkidesse jagada ja plokid pakettidesse jagada (*interleaving*). Vastuvõtja järjestab saabunud pakettidest plokid õigesse ajalise järjekorda. Paketi kao korral on palju lühikesi helikatkestusi ühe pikema asemel.

Vastuvõtjas on realiseeritud kao peitmise skeemid:

- vaikuse asendamine (kõige levinum ja lihtsam, efektiivne kui paketid on lühikesed (<4ms) ja väiksel kaol (<2%)),
- paketi kordamine (kadunud paketi asendatakse talle eelnenud paketi koopiaga – väike arvutuskeerukus),

- mustri jälgimise parandus (kasutab kadunud pakatile eelnevat ja järgnevat paketti ja interpoleerib nendest sobiva signaali kao peitmiseks – kvaliteedilt hea, aga arvutuslikult keeruline).

Pakub parimat võrgutingimustele vastavat helikvaliteeti. [30]

1.6.2.4 wbd

Töö ülesande täitmisel vähe kasutatud, aga muidu täiesti kasutatav ja kasulik programm.

Wbd on jagatud töölaud videokonverentsis, mitte lihtsalt joonistusvahend. Iga kasutaja saab joonistada ja kirjutada. Kasutaja ei saa teise kasutaja teksti või joonistust kustutada (peab selle peale joonistama või kirjutama). Kasutaja ei saa teiste kirjutatud teksti muuta. Tekst ja joonised on värvilised, on võimalik lisada ka faile, mis aga peavad olema PostScript formaadis. [32]

1.6.2.5 ReLaTe

ReLaTe (Remote Language Teaching) [24] on õppetarkvara, mis ühendab rat, vic, wbd ja nte (Network Text Editor) [19] ja võimaldab õpetada. Kuni 8 osalejat saavad näha (kui saadab videot) ja kuulda üksteist, kasutada ühist valget tahvlit ja ühiselt teksti muuta. (*ReLaTe is a joint project between University College London and Exeter University.*)

1.6.2.6 Kokkuvõtteks.

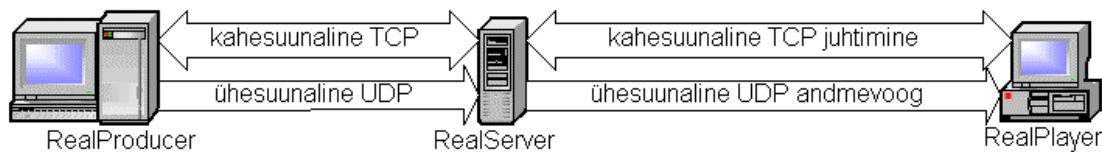
Mbone tools on väga hea tarkvara interaktiivseks multimeedia edastamiseks ning kasutab ära multisaate võimalusi. Kasutaja saab palju parameetreid ise seadistada. Nii on näiteks vic programmi kasutades võimalik näha, milliseks muutub kujutis, kui muuta võrgu ja video kodeerimise-pakkimise parameetreid.

Kahjuks ei saa saadetavat või vastuvõetavat multimeediat salvestada. (rat uuem versioon 4.2.4 võimaldab vastuvõetavat heli salvestada faili ja failist heli saata. Vic saab saata varem viciga salvestatud videot failist). Kasutajale võib olla ebamugav käsurea parameetreid seada, aga kui sessioon sdr kaudu käivitada, siis ei peaks see probleem olema. Multisaate piiratud levi tõttu on sessioonis osalejate ring piiratud. Erinevate võrgutingimustega kasutajatele edastatakse samade parameetritega multimeediat. Et kõik osalejad saaks vajaliku kvaliteediga multimeediat, tuleb selle parameetrid seada kõige halvemate võrgutingimustega osaleja järgi.

1.6.3 Real Networks tarkvara

Kommertstarkvara (Real Networks Inc, <http://www.real.com>), mille osasid versioone (*Basic*) on võimalik tasuta kasutada. Tarkvara pere, mis kasutab RTSP protokollide võimalusi, õigemini RTSP protokoll tahetakse standardiks saada, kuna see levinud tarkvara kasutab seda juba.

Kasutajatele saab anda suure mugavuse. Vaid ühe programmiga saab vastu võtta ja ette mängida videot koos heliga ning sellele lisatud sündmusi nagu näiteks veebilehe avamine veebibrauseris mingil ajahetkel. Kasutaja pääseb salvestatud sessioonile ligi vaid ühe hiireklõpsuga õigel veebilehel ja RealPlayer võtab ise üle. Ettemängimiseks pole vaja oodata kogu klipi saabumist võrgust, vaid saab juba alustada, kui puhvritäis on kohal ning ettemängimise ajal saabub ülejäänud.



Joonis 10. Real Networks programmide omavaheline koostöö [23]

1.6.3.1 RealPlayer

Multimeedia sessioonide vastuvõtu ja ettemängimise tarkvara. Võimalik ette mängida nii salvestatud multimeedia klippe kui ka reaajas toimuvaid ülekandeid. Vastavalt oma internetiühenduse kiirusele on võimalik valida klipi allalaadimise edastuskiirust (kui klipp on mitme edastuskiirusega salvestatud st *SureStream*) Kasutaja saab seada ka vastuvõtuks kasutatavat transpordiprotokolli (TCP, UDP või HTTP). Programm võib olla ka veebibrauseri lisandprogramm (*plugin*), mis annab juurde kasutusmugavust.

1.6.3.2 RealServer

Multimeedia sessioonide saatmise ja teenindamise tarkvara. Kliendid ühenduvad selle serveriga mõlemat tüüpi sessioonides (reaajas toimuvate ja nn simuleeritud reaalarja) osalemiseks. Võimaldab kliendile pakkuda vastavalt ühenduse loomisel kokkulepitud parameetritele andmeedastuskiirust ja transpordiprotokolli ning ka multisaadet.

Server pakub oma kettaruumi reaalarja ülekannete arhiveerimiseks, et neid hiljem simuleeritud reaalarjas kasutajatele saata ettemängimiseks. Samuti saab kasutajatele saata ettemängimiseks igal muul viisil saadud RealMedia faile.

RealServer Basic 8.0.1 arhiveeritud failidel on puhverdamise aja parameetri väärtus vale (puhverdamise aeg peaaegu kogu klipi pikkune), kui seda klippi vaadatakse võrgust (kasutades RTSP protokolli). Kui sama klippi vaadatakse kettalt, siis on puhverdamise aeg piisavalt lühike.

1.6.3.3 RealProducer

Multimeedia sessioonide loomise ja salvestamise tarkvara. Reaalarjas toimuva sessiooni andmed saadab RealProducer Realserverile, kust RealPlayerid neid kätte saavad. Video ja heli on võimalik salvestada ühte faili mitme erineva andmeedastuskiirusega ja vastavalt kiirustele ka erineva pakkimistihedusega koodekitega.

RealProducer soovib, et videohõive kaart annaks talle YUV andmeid: I420, IYUV, YV12, planar YUV 4:2:0, YUY2, UYVY, packed YUV 4:2:2, YVU9, RGB24. [RealProducer veateade].

RealProduceri Windows versiooni kasutamisel arvuti elektrisäästu süsteem ei tohiks olla sees, st *Power scheme* peaks olema *Always on* koos kõikide parameetrite väärtustega *Never* (*System standby*, *Turn off monitor* ja *Turn off hard disks*).

1.6.3.4 RealPresenter

Tarkvara, millega saab Microsoft PowerPoint ettekandeid koos juurdekuuluva heli ja videoga võrku panna. Samuti nii reaalarjas kui simuleeritud reaalarjas. Vaatajatele aitab

RealPlayerist, mille video aknas näidatakse slaidi jpg faili, väike videoekraan ja slaidide sisukord, mis on klikitav.

Kaasa installeeritakse ka RealServer, mis jagab mõlemat tüüpi presentatsioone võrku. Baasversioon kasutabki ainult samas arvutis töötavat RealServerit, *Plus* versioon oskab ka teises arvutis asuvat RealServerit kasutada.

Loengu pidaja arvutis on avatud MS PowerPoint ja selles RealPresenter, kus algav sessioon ära kirjeldatud. Reaalajas vaatajad peavad teadma (veebilehe) aadressi, kus on link sessioonile, mida RealPlayer hakkab näitama. PowerPoint slaidid vahetuvad, siis kui loengu pidaja slide vahetab. Vaatajad näevad seda muidugi sünkroonis heli ja pildiga.

Tähelepanelik peab olema installeerides, sest RealPresenter Basic 8.01 ei tööta MS Office 2002 ja XP versioonidega ja Windows XP-ga.

1.6.3.5 Kokkuvõtteks.

RealMedia tarkvara on väga mugav veebi kaudu juurdepääsetavate multimeedia failide jagamiseks. Samuti saab jagada reaalajas toimuvat multimeedia sessiooni. Olenevalt kasutaja võrgutingimustest saab server saata talle erineva kiirusega andmevoo teisi kasutajaid mõjutamata. Multimeediat saab ka sünkroniseerida näiteks veebilehtedega (teatud hetkel laetakse veebibrauserisse mingi veebileht).

Puuduseks võib pidada seda, et sessiooni hoidja ei saa tagasisidet sessiooni sisu kohta.

Lähteülesande täitmisel on siiani osutunud sobivaimaks RealNetworks süsteem koosnevana RealProducerist ja RealServerist ning vaatajatele RealPlayer. Kuid siin osutus suurimaks probleemiks õppematerjalide ja loengu klipide vaheline sünkroonsus. Pärast saab salvestatud klipis määrata iga slaidi näitamise ajahetked ja siis vastavat veebilehte näidata veebibrauseris. Kuid selleks on olemas ka tarkvara RealPresenter, mis ise sünkroniseerib slaidid ning loengu heli ja video. Kuid RealPresenter tasuta versioon ei saa kasutada teises arvutis asuvat RealServerit

1.7 Teiste kogemused

Internetis on varemgi muude õppematerjalide kõrval ka loengu multimeedia salvestust näidatud. Side loengu tegemisel oli suureks eeskujuks Kalifornia ülikooli töö. Eestiski on loenguid interneti kaudu üle kantud.

Arvatavasti kasutatakse seda teenust firmade sisevõrkudes rohkem, sest multimeedia salvestuste ettevalmistamine on rahaliselt ja ajaliselt kulukas. Vaatleme avalikke teenuseid.

1.7.1 Kalifornia ülikoolis toimuvad loengud

Kalifornia ülikool Berkeley valib mõned kursused (videohõive võimaluste, vajaduse ja multimeedia salvestuste tootmise võimaluste järgi), mille loenguid saab reaalajas ja hiljem loengute salvestusi vaadata internetist. Süsteemi nimetatakse webcast@berkeley. Õpetamise tehnoloogia keskus (*Educational Technology Service*, <http://media.berkeley.edu>) pakub heli ja videohõive teenust ning infoteenuste ja tehnoloogia keskus (*Information Systems and Technology*, <http://ist.berkeley.edu>) hooldab *webcast* andmebaasi ning meediavoo teenuse ja hostingu infrastruktuuri. Selline teenus loodi Berkeley multimeedia uurimiskeskuses (*Berkeley Multimedia Research Center* (BMRC), <http://bmrc.berkeley.edu>). Kursuse kodulehel on lingid loengute videotele, mida saab vaadata iga arvutiga, millel on RealPlayer.

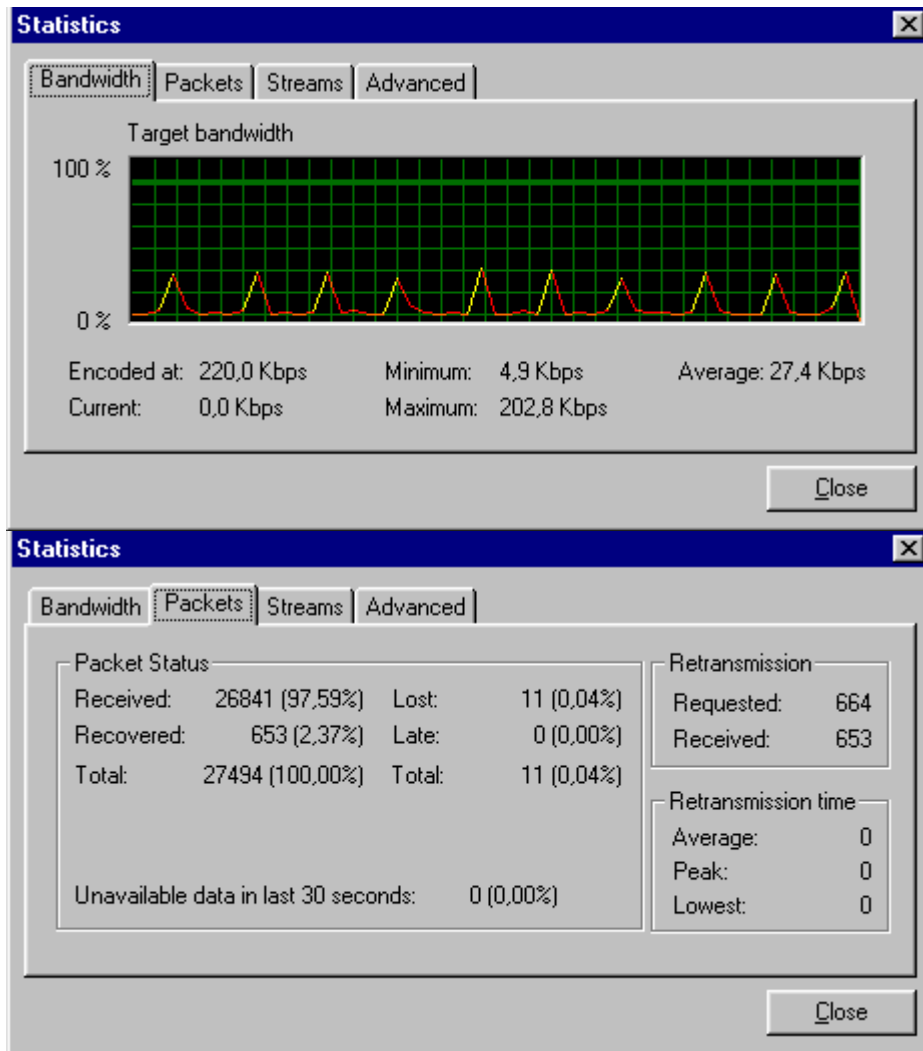
Praegu kasutatakse firma Real Networks tehnoloogiat ja tarkvara. Kasutajal (kuulajal-vaatajal) peab olema veebibrauser (Netscape või Internet Explorer) ja RealPlayer. Kasutaja interneti kiiruseks soovitatakse vähemalt 56k modem (miinimum) ja sellega on kujutise kvaliteet kehv. Kasutajate ring pole piiratud, loenguid saab kasutada nii tolle ülikooli arvutivõrgust, Ameerika internetist kui ka Eestist.

On võimalik vaadata reaajas toimuvaid *webcaste* nende tegelikul toimumise ajal ja selle arhiiv tekib sama päeva jooksul. Kui aga oli reaajas hõivega (*capture*) probleem, siis arhiivi loomine videolindilt võtab aega kuni 3 päeva. Arhiive hoitakse alles 1 aasta. Kasutaja ei saa omale salvestada loengute arhiive autoriõigustest tulevate piirangute tõttu. Interaktiivsust ei ole. [37]

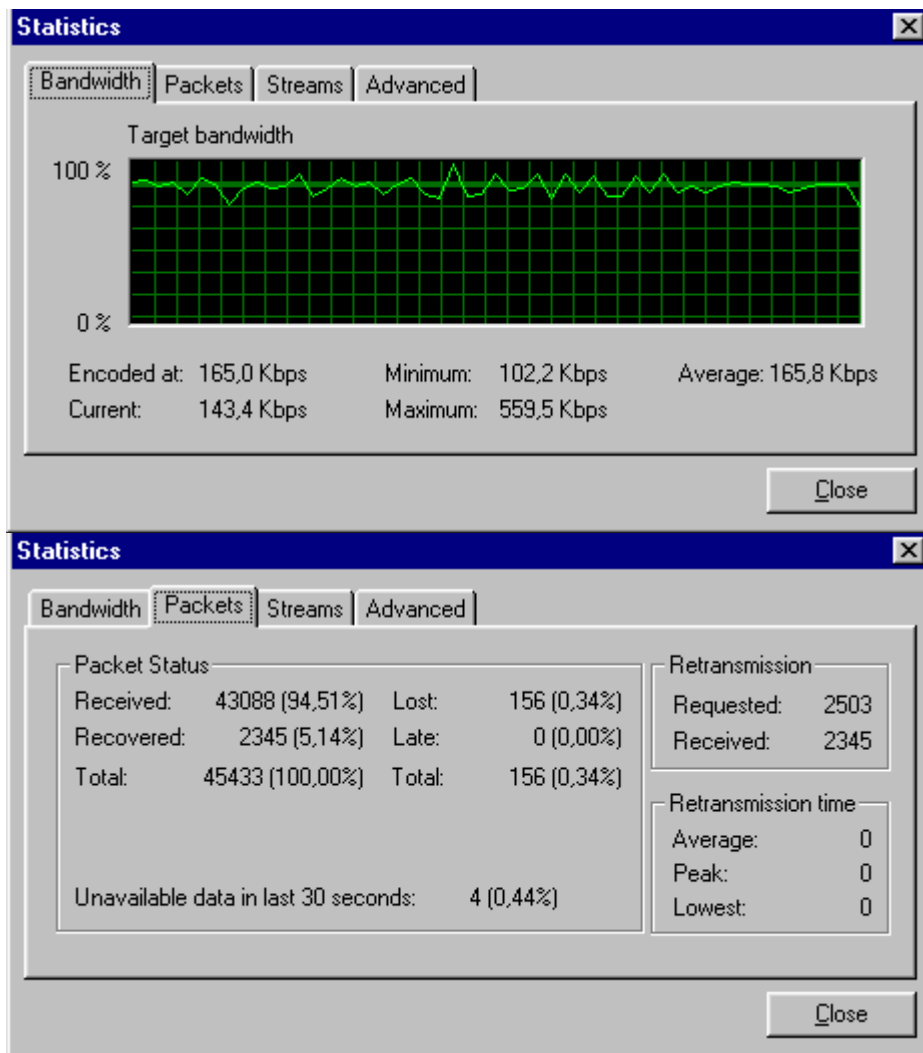
Kuulata saab heli (mikrofon on õppejõu juures ja mõned ka saalis kuulajatele küsimuste esitamiseks), vaadata saab pilti - 1 kaamera õppejõu jälgimiseks, 1 saali üldvaade, loengu abimaterjalide jälgimiseks – need kordamööda lülitatakse väljundisse, mida vaatajad näevad.

Näiteks õppeaine *Intro to computers* loeng *Future of computing* reedel, 30. novembril kell 4.00-5.00 PST (kohaliku aja järgi). Stream0 on audio, kodek RealAudio 5.0, 16 kb/s kõnekodek. Stream1 on video, Realvideo G2 kodek, kujutise suurus 320x240, kaadrisagedus 15 ja 30 kaadrit sekundis, andmeedastuskiirused 34,0 kb/s, 21,5 kb/s, 14,0 kb/s, 204,0 kb/s, 149,0 kb/s ja 34,0 kb/s. RealPlayer saab valida 5 erineva kodeeritud andmevoo kiirusega teenust (heli kiirus + video kiirus): 16,0+14,0=30 kb/s, 16,0+21,5=37,5 kb/s, 16,0+34,0=50 kb/s, 16,0+149,0=165 kb/s ja 16,0+204,0=220 kb/s. Loeng kestab 49 min. Keskmine saadud edastuskiirus oli 180 kb/s. RealServer Version 8.0.1.367 (sunos-5.7-sparc), transpordiprotokoll UDP.

Siin mõned RealPlayer statistika pildid (Joonis 11, Joonis 12 ja Joonis 13).



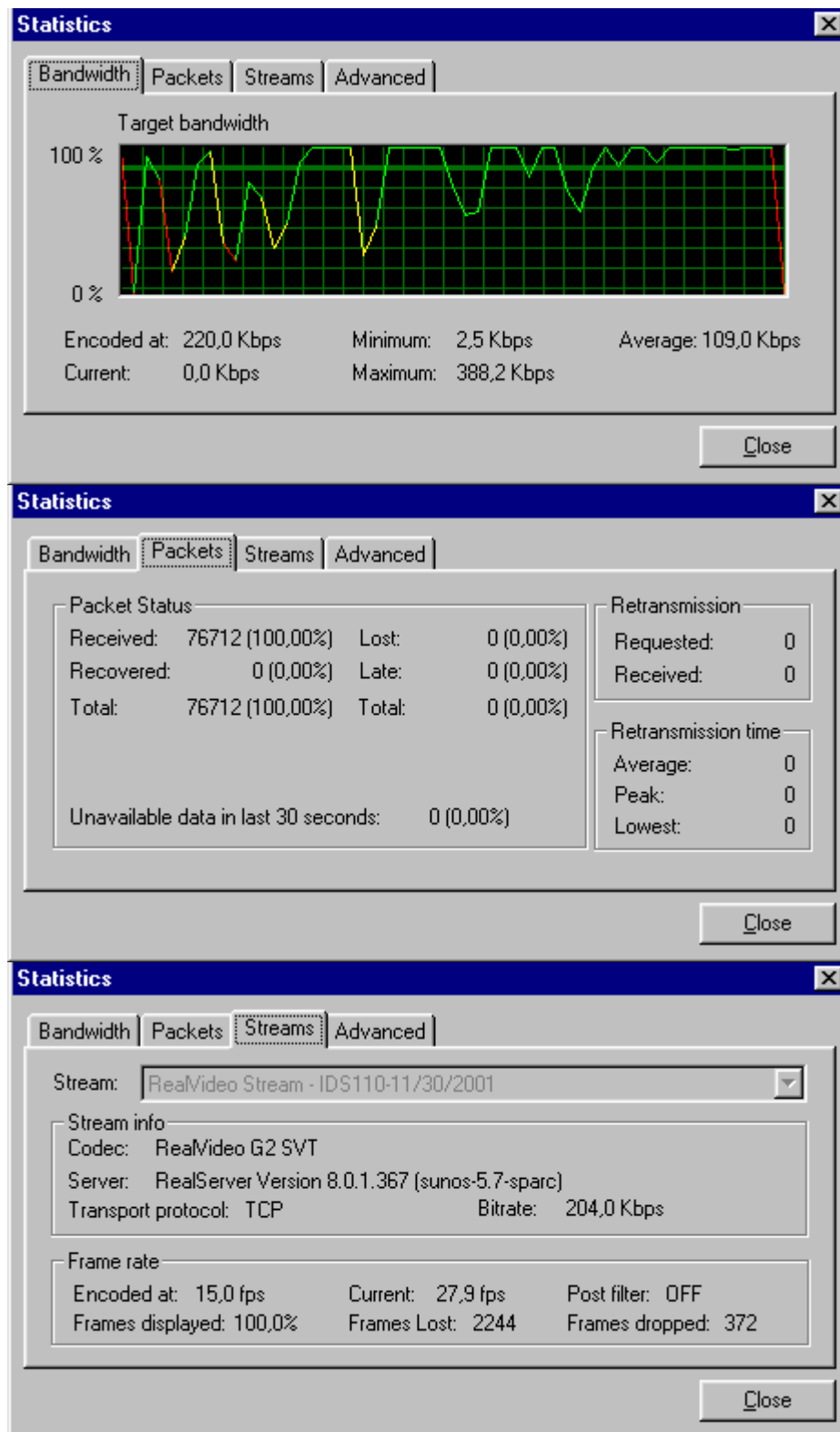
Joonis 11. RealPlayer statistika



Joonis 12. RealPlayer statistika

TCP.

50 kb/s, 165 kb/s, 37,5 kb/s, 220 kb/s.



Joonis 13. RealPlayer statistika

1.7.2 BIBS

BIBS (*Berkeley Internet Broadcasting System*) on interaktiivne televisioonisüsteem kaugõppe, hajustötamise ja interaktiivse televisiooni rakenduste jaoks Kalifornia ülikooli Berkeley linnakus. [3] Töötas kuni 2001 sügis. Kasutati enne, kui loodi uuem keskkond, millest oli eelmises peatükis.

1998 sügissemestril alustas pärast 3-a katsete ja testimise aega Berkeley multimeedia, liideste ja graafika seminar (<http://www.bmrc.berkeley.edu/courseware/cs298>). Tarkvarast kasutatakse kas RealPlayer G2 (ver 5,6,7, praegune on 8) või Internet Mbone (kasutab multisaadet) tarkvaraga. On võimalik kasutada Macintoshiga Quicktime4 Playerit. Vaatajate hulk on piiratud Mbone levikuga ja ülikooli arvutitesse installeeritud tarkvaraga. Võrgu kiirus on 50-500 kb/s. Real/G2 programmiga on võimalik vaadata alates 28,8 kb/s väga halva pildikvaliteediga ja soovitatakse võrgu edastuskiirust alates 50 kb/s. Mbone vajab võrgu kiirust 100-800 kb/s. [4]

Interneti kaudu osalejatel küsimuste esitamise võimaluseks tehakse katseid programmiga QuestionBoard (1997, november, <http://bmrc.berkeley.edu/papers/1997/137/137.html>). QuestionBoard qb võimaldab Mbone sessioonis osalejatel esitada küsimusi ja sessiooni moderaatoril juhtida sessiooni õppejõudu häirimata. Küsimused esitatakse tekstisõnumina programmis qb või kui sessiooni moderaator on küsimuse soovist teada saanud ja õppejõud ka, siis saab küsimuse audios esitada.

1.7.3 Lecture Browser

LB (*Lecture Browser*) on Javascriptis kirjutatud veebibrauseri põhine töövahend. Ta ühendab videovoo ja sünkroniseeritud slaidid.

Erinevad meediaelemendid hoitakse andmebaasis ja kasutajad pääsevad neile ligi standardsete HTTP päringutega. Kasutajad saavad teostada otsingut tekstist ja meediast (slaidid ja multimeedia) üksteisest sõltumatult.

LB algne kood on saadud Cornelli ülikoolist (*Cornell University*). BMRC arendab seda edasi lisades audio otsingu, *bookmarks* ja *online* üliõpilastele interaktiivse valge tahvli. [5]

1.7.4 Kogemused Eestis

Videokonverentsi ja loengutest multimeedia salvestuste interneti kaudu edastamisel on ka Eestis vähemalt katsetusi tehtud.

Allikas [35] seletab avatud õppe ja kaugkoolituse mõistet. Selle sees on võimalik õppematerjale õppijateni tuua videokonverentsi või video nõudmisel meetodil.

Eestis tegeldakse virtuaalõppega ja kasutatakse WebCT programmi näiteks IT kolledžis [36] ja Tartu Ülikoolis [33]. See programm WebCT võimaldab õppeainega seotud õppematerjale veebilehele panna, kasutada suhtlemisvahendeid – foorumit, postkasti, jututuba, kasutada kalendrit ja koostada ja kasutada teadmiste kontrollimiseks teste. [33]

WebCT on veebi kaudu pakutav keskkond, kus antakse õppijale kõiksuguseid õppematerjale ja õppija saab seal ka testidega oma teadmisi kontrollida. [38]

IT kolledžis on 1 auditooriumite paar audio ja videotehnikaga virtuaalauditooriumiks. [36]

Videokonverentsi katsetusi on olemas ülikoolidel. IT kolledži avalikud loengud teostatakse RealNetworks tarkvaraga [2] Videokonverentsiga tegeleb firma Datel [34].

1.8 Kokkuvõte

Tööülesanne on loeng internetti panna, siis esimeses peatükis on vaadeldud selle töö kõiki osasid: heli ja video digitaliseerimine, saadud andmete arvutivõrgu kaudu edastamine ja arvutis ettemängimine. Lühidalt on kirjeldatud selleks vajalikku

riistvara ja tarkvara, kuidas analoogsignaale arvutisse saada, ning vaadeldud tarkvara multimeedia andmevoo edastamiseks internetis.

Töö esimeses alapeatükis on lühidalt vaadeldud multimeedia taju iseärasusi, mida kasutatakse ära multimeedia digitaliseerimise kvaliteedi määramisel. See võimaldab saavutada andmemahu vähenemist koos video pildi kvaliteedi mittetajutava vähendamisega. Teises ja kolmandas alapeatükis on kirjeldatud meetodeid ja algoritme, kuidas heli ja videosignaal digitaliseeritakse. Analoogsignaal diskreeditakse (ajas) ja kvanditakse (amplituudis). Saadud andmehulk on väga suur kohe võrgus edastamiseks ning see kodeeritakse ja pakitakse koodekiga ja edastatakse võrku ja/või vajadusel salvestatakse faili. Ettemängimisel andmevoog pakitakse lahti, dekodeeritakse ja saadud väärtustest taastatakse heli- ja videosignaal. Neljas alapeatükk käsitleb multimeedia edastuse protokolle ja süsteeme. Viiendas alapeatükis on kirjeldatud riistvara – helikaart ja videohõive kaart, mille abil analoogsignaal arvutisse viiakse, ja on antud edasi kogemusi, mis tekkisid selle riistvara kasutamisel. Kuuendas peatükis on vaadeldud erinevat tarkvara, mida kasutati loengu interneti ülekandmisel erinevatel aegadel side loengutes ning on sellele tarkvarale antud ka hinnang tööülesande täitmiseks sobivuse seisukohalt. Tarkvara moodustas tavaliselt nn programmide pere, kus on olemas programmid multimeedia andmevoo või faili loomiseks, edastamiseks, vastuvõtuks ja ettemängimiseks. Kõik tarkvara peresse kuuluvad programmid ühilduvad oma peresse kuuluvate programmidega, aga mitte teise peresse kuuluvate programmidega. Kasutamisel eelistatavamateks saavad arvatavasti need tarkvarade pered, mille kasutamine on lihtne ja mida on internetist lihtsam kätte saada või on lihtsalt kõige levinuma kasutusega. Seitsmes alapeatükk oli nendest kogemustest, mida Eestis ja maailmas on saadud multimeedia levitamisel interneti kaudu.

2 Multimeedia ülekanne piiratud ressursside puhul

Testime, milliste parameetritega sidekanal on sobiv. Kuidas muutuvad võrguparameetrid rakenduse poolt pakutava teenuse kvaliteeti mõjutavad? Kas RealPlayeri statistika annab kasutajale õige ettekujutuse võrgu olukorrast? Millist transpordiprotokolli (TCP, HTTP või UDP) on kõige parem kasutada erinevate maksustamise viiside korral?

2.1 Kirjeldus

Teostati katsed multimeedia faili edastuseks erinevatel võrgutingimustel. Iga parameetrite komplekti korral teostati 2-5 katset. Väiksem katsete arv teostati, kui järjestikuste katsete tulemused tulid lähedased, suurem katsete arv teostati, kui järjestikuste katsete tulemused erinesid palju või kliipi vaatamine ebaõnnestus (kliendi ja serveri vahelise ühenduse katkemine).

2.1.1 Riistvara

Katse riistvara koosneb 3 arvutist, mis on omavahel võrku ühendatud. Ühes arvutis (server) töötab RealServer, mille kõvakettal asub multimeedia fail. Teises arvutis (klient) töötab RealPlayer, millega seda faili vaadatakse. Kolmas arvuti (gate) nende kahe vahel on arvuti, mis on marsruuteriks teiste vahel ja millel töötab ka võrgu simuleerimise programm MATE (*Mobile Application Test Environment*)



Joonis 14. testimisel kasutatud testvõrk

Testimisel kasutatud arvutid:

- Server: Pentium III, 1 GHz, mälu 512 MB, kõvaketas 2x45 GB, võrgukaart Intel.
- Klient: Pentium III, 550 MHz, mälu 256 MB, kõvaketas 2x20 GB, helikaart SoundBlaster Live, videokaart ATI RADEON, võrgukaart Intel.
- Gate: Pentium III, 550 MHz, mälu 256 MB, kõvaketas 2x20 GB, võrgukaart 2x 3Com.

2.1.2 Tarkvara

Testvõrgu arvutites kasutati järgmist tarkvara:

- Server: RealServer Basic Version 8.0.1.367 (win32), Windows 2000 Server.
- Klient: RealPlayer Basic 8 (win32) Build 6.0.9.584, Windows 98.
- Gate: MATE version 2.2.5, Epan v1.3.1, Linux RedHat 6.1.

MATE (*Mobile Application Test Environment*) on mobiilse pakettvõrgu simulaator, mida kasutatakse rakenduse (arvutiprogrammide) testimiseks ja mobiilse andmeedastuse keskkonna demonstreerimiseks. Testitakse, kuidas rakendus tuleb toime erinevate võrgutingimuste korral. Erinevate võrguparameetrite määramisega saab simuleerida erinevaid võrgutingimusi. Võrguparameetreid (latentsus,

läbilaskevõime, paketikadu, bitivea tõenäosus ja võrgukatkestus) saab muuta ja seda eraldi võrguliikuse mõlemate suundade jaoks (kliendist serverisse st üleslülili ja serverist klienti st allalülili). [16] Latentsus tähendab rakenduse aeglast reageerimist kasutajale, kuna pakettide edastamisel võrgus tekib viide. Läbilaskevõime on edastuskiirus, kui kiiresti saab võrgus andmeid edastada. Paketikadu tähendab seda, et osa pakettidest ei jõua saatjast vastuvõtjani, st kaovad. Bitivea tõenäosus tähendab tõenäosust, et biti väärtuse asemel on selle biti pöördväärtus, simulaator tekitab bitivea paketi andmete ossa, mitte päistesse. Võrgukatkestus tähendab, et saatja ja vastuvõtja (kliendi ja serveri) vahel ei saa liikuda ükski pakett. Võrgukatkestust testimisel ei kasutata.

Epan (*Ethernet Protocol Analyzer*) on võrgu pakettide püüdmise ja analüüsimise programm koos graafilise kasutajaliidesega. Võimaldab kogutud pakettide kohta statistikat saada ja dekodeerib paketi päised. [9]

Kliendi ja serveri vaheline andmevahetus juhitakse läbi arvuti gate, kus on võrguparameetrid paika seatud. Kliendi arvutis saab vaadata, kuidas rakendus käitub selliste võrguparameetrite korral.

Testimiseks valiti, kuidas MATE kaudu muudetavatest võrguparameetritest igaüks eraldi mõjutab saadava teenuse kvaliteeti. Kvaliteetseks loeti teenust siis, kui ettemängimisel katkestusi ei tekkinud.

2.1.3 Seaded

Testitav fail on Side loeng. Loengu faile on võimalik serverist edastada 2 erineval kodeeritud andmevoo kiirusel: 20 kb/s ja 350 kb/s. Valitud on 2-minutiline osa ühe loengu algusest. Täpselt 2-minutilise osa saamiseks on tehtud fail test.ram, mis sisaldab järgmist teksti:

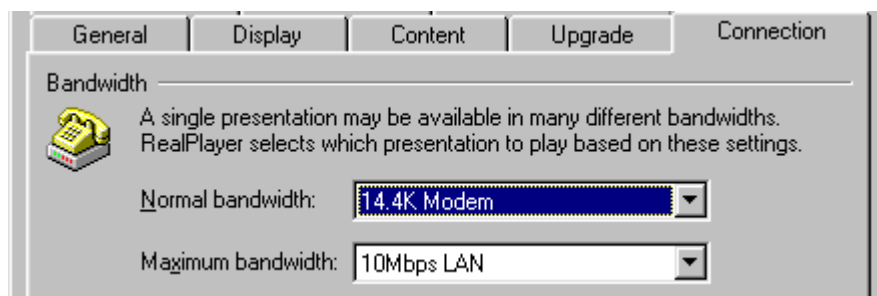
```
rtsp://193.40.252.41:554/side/Side_20011122_osa1.rm?end="00:00:02:00,00"
```

Reaalaja rakendus ei vaja kogu klippi, et alustada selle ettemängimist, vaid võib juba alustada, kui on saabunud puhvritäis (kuni 30 s) ja ettemängimise ajal vastu võtta klippi ülejäänud andmeid.

Uurime, kuidas see läbi mobiilvõrgu kasutajani jõuab. Mobiilvõrgu (GPRS) edastuskiirus on väike, latentsus suur. Loengu fail on kodeeritud 2 erineval edastuskiirusel. Selleks et RealPlayer valiks väiksema kiirusega andmevoo, on seaded:

View->Preferences->Connection: Normal Bandwidth 14.4K Modem.

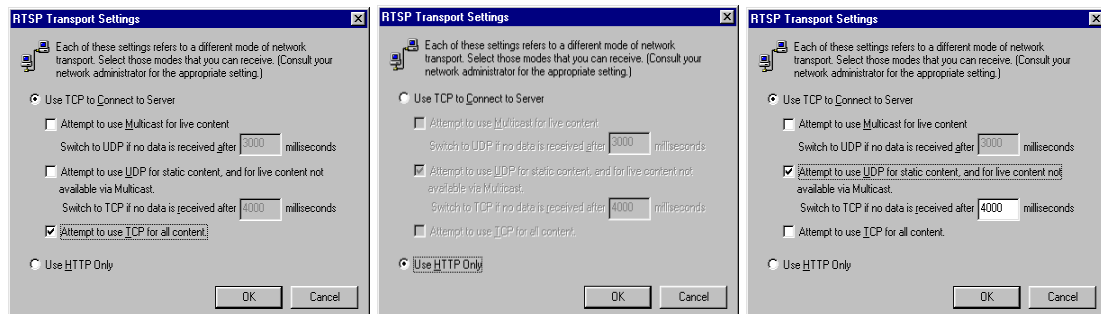
Et juhuks, kui võrk on tegelikult parem, siis valime seal *Maximum bandwidth 10Mbps LAN*.



Joonis 15. RealPlayeris seatud oma võrgu läbilaskevõime parameetrid

RealPlayeris valime kasutatava transpordiprotokolli menüüst View->Preferences->Transport->RTSP settings.

RealPlayer loob ühenduse serveriga kasutades TCP protokollit. Klipi andmevoogu on ta võimeline vastu võtma kas UDP, TCP või HTTP protokollit kasutades (Joonis 16). HTTP on eelkõige selleks, et tulemüürist andmevoog läbi läheks, töötab TCP peal. UDP on soovitatav kasutada. Kui aga UDP-ga ei tööta, siis abiinfo soovitab TCP-d. Testides kasutame klipi andmevoogu edastamiseks kõiki protokolle, aga korraga neist ühte protokollit klipi vaatamise jooksul.



Joonis 16. Vasakult alates: TCP, HTTP ja UDP on valitud transpordiprotokolliks

2.1.4 Töö käik

Programmis MATE seatakse vajalikud võrgu parameetrid. RealPlayeris seatakse vajalikud seaded. Nullitakse stopper. Kontrollitakse pingi aega ja märgitakse tulemus üles (kui vaja).

Siis alustatakse Epaniga andmete kogumist. Klientarvuti RealPlayeris valitakse ettemängiseks fail `test.ram`. Tehakse hiireklõps nupul PLAY ja sellel hetkel alustatakse stopperiga aja võtmist. Kui RealPlayeri aknas on näha esimene kaader, siis võetakse vaheaeg. Edasi jälgitakse RealPlayeri statistika aknast, kui enam uusi pakette ei saabu (graafik ajas edasi ei liigu), siis võetakse järgmine (teine) vaheaeg. Kui faili ettemängimine on lõpetatud, siis võetakse lõplik aeg. Lõpetatakse Epani pakettide kogumine.

Kui ettemängimisel oli katkestusi puhverdamise tõttu, siis märgitakse see üles. Kui oli muid huvitavaid sündmusi, siis märgitakse need samuti üles. Märgitakse üles Epani kogutud andmed, RealPlayeri kogutud andmed (statistika) ja stopperiga mõõdetud ajad.

Korrata kuni kõik katsed tehtud.

2.1.5 Kogutavad andmed

RealPlayeril on oma statistika (menüüst *View->Statistics*). Sealt saab näha andmeedastuskiirust (min, max, keskmine), pakettide arvu (vastuvõetud, taastatud, kaotatud, uuesti saadetud), kaadrisagedust video kohta, kaotatud ja välja jäetud kaadrite arvu.

Epan pakub samuti statistikat. Valiti sealt järgmised kogutavad andmed: keskmine paketi pikkus, keskmine võrgu (10 Mb/s) koormus, keskmine bitikiirus, keskmine paketedastuskiirus.

Programmi MATE kasutati ka üleslüli ja allalüli pakettide arvu ja andmehulga mõõtmiseks.

Stopperiga mõõdetakse aega kasutaja hiireklõpsust nupule PLAY kuni esimese pildi nägemiseni, klipi andmete edastuse lõpu aega ja kogu aega, mis kulus 2-minutilise multimeediaklipi vaatamiseks. Aja mõõtmise täpsus on kümnendik sekundit, millele lisandub mõõtmisviga, mis tuleb inimese reaktsiooni kiirusest toimuva sündmuse fikseerimisel ja seda võib hinnata umbes sekundi suuruseks.

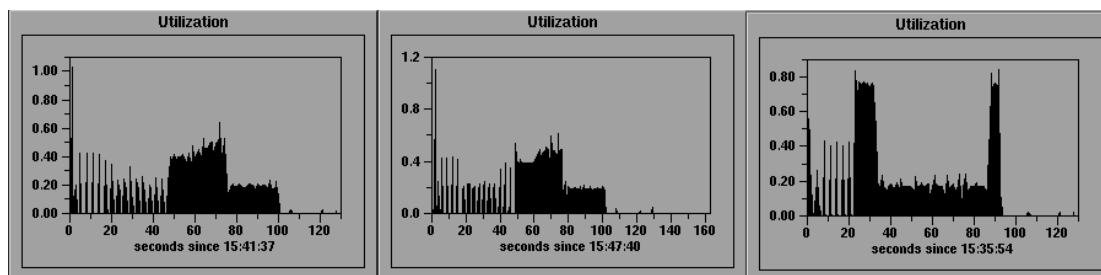
Mõõdetud aja järgi saab teada, kui palju maksab 2-minutilise multimeediaklipi vaatamine, kui võrgukasutust maksustatakse aja eest. Mõõdetud andmehulga järgi saab teada, kui palju maksab 2-minutilise multimeediaklipi vaatamine, kui võrgukasutust maksustatakse andmehulga eest. Epani statistika annab andmed pakettide kohta, mis läbisid võrku, üleslülili ja allalülili kokku. RealPlayer annab rakendustaseme statistika andmepakettide kohta.

RealPlayeri statistika näitab seda, mida kasutaja saab teada ja mille põhjal ta võib teha järeldusi võrgu olukorra kohta. Kasutaja tavaliselt ei tea võrgu olukorrast, ta ei saa näha Epani ega Mate tulemusi, aga soovi korral saab näha RealPlayeri statistikat klipi vaatamise ajal.

2.2 Andmevoo profiil piiranguteta võrguga

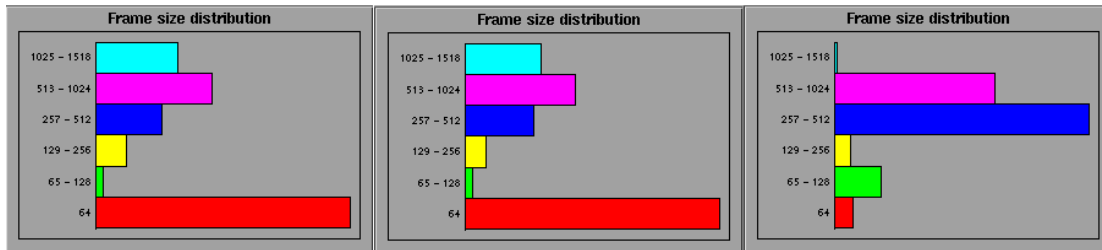
Testitud on 3 edastusprotokollit TCP, HTTP ja UDP. HTTP ja TCP profiilid on sarnased üksteisele, UDP on erinev. Testides erinevate võrgutingimustega oli TCP ja HTTP käitumine sarnane.

2.2.1 Andmevoo edastus



Joonis 17. TCP, HTTP ja UDP paketi voog ajas.

Jooniselt (Joonis 17) on näha, et alguses toimub ühenduse loomine, siis server saadab iga paari sekundi järel väikse hulga 40 kb andmeid. Samal ajal RealPlayer näitas kodeeritud andmevoo kiiruseks (*Encoding at*) 292,4 kb/s ja videot näitas kaadrisagedusega 0,5 kaadrit iga 2 sekundi järel. Siis TCP ja HTTP korral 45 sekundi pärast ja UDP korral juba 22 sekundi pärast lülitatakse ümber kodeeritud kiirusele 20 kb/s ja server hakkab andmeid saatma iga sekund umbes 40 kb, UDP korral aga iga sekund 70-80 kb. Siis, arvatavasti, kui vastuvõtjal RealPlayeril pole andmeid kuskile salvestada ja puhver täis, siis edastatakse andmeid kõikidel juhtudel 20 kb/s kuni klipi lõpuni. Välja arvatud UDP korral kui viimased 10 sekundit edastatakse andmeid jälle 80 kb/s. Kui andmed saadetakse, siis varsti lõpetatakse ühendus, see võib juhtuda ka enne ettemängimise lõpetamist.



Joonis 18. TCP, HTTP ja UDP pakettide jaotus suuruse järgi.

Paketi suuruslik jaotus erineb TCP, HTTP ja UDP vahel. TCP ja HTTP korral on kõige rohkem pakette pikkusega 64 B, UDP korral aga pikkusega 257-512 B. UDP korral on kõige vähem pakette pikkusega 1025-1518 B, aga TCP ja HTTP korral hoopiski kõige vähem pakette pikkusega 65-128 B. See kinnitab seda, et TCP ja HTTP korral saadetakse palju kinnituse teateid paketi kättesaamise kohta.

2.3 Interpreteeritud andmed

Siin esitatakse testitulemuste andmed ja nende analüüs. Uuritakse, kuidas võrguparameetrid – latentsus, läbilaskevõime, paketikadu ja bitivea tõenäosus – nii üleslulis kui allalulis mõjutavad klipi vaatamiseks kulunud aja pikkust, ettemängimisel katkestuste tekkimist (puhvri tühjenemise tõttu), andmehulga suurst, võrgus liikunud pakettide arvu ja RealPlayeri statistikat. Piiratud võrguparameetrite korral klipi ettemängimise ajal kogutud testitulemusi võrreldakse nendega, mis saadud piiramata võrguparameetrite korral.

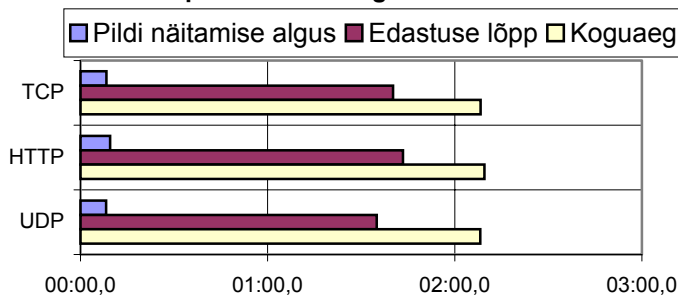
Bitivea tõenäosused on tähistatud $1e-9$, $1e-6$, $1e-5$ ja $1e-4$, mis tähendavad vastavalt 10^{-9} , 10^{-6} , 10^{-5} ja 10^{-4} .

2.3.1 Piiramata võrgutingimuste korral saadud testitulemused

Aeg

Mõõdeti aega kasutaja hiireklõpsust RealPlayeri nupule PLAY kuni klippi ettemängimise alustamiseni (Pildi näitamise algus), andmevoo edastuse lõpuni (Edastuse lõpp) ja ettemängimise lõpuni (kogu-aeg).

2-minutilise klipi vaatamise aeg

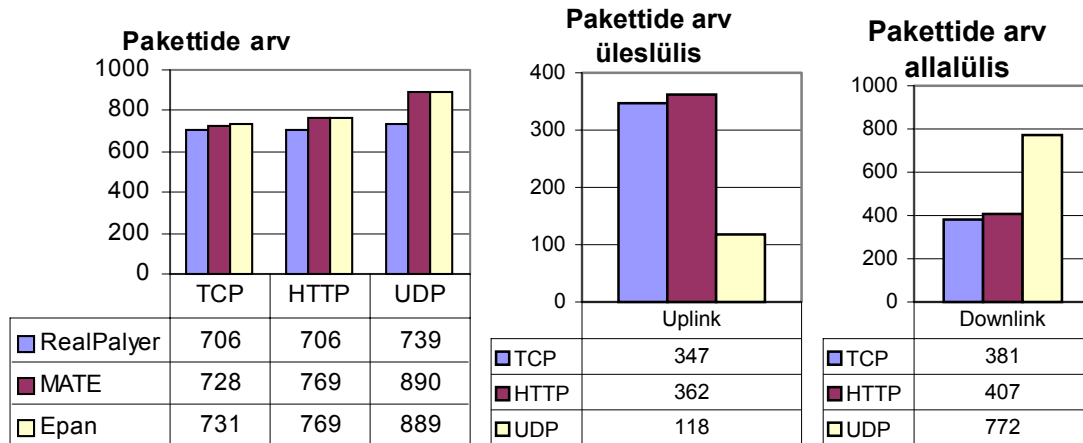


Protokoll	Pildi näitamise algus	Edastuse lõpp	Kogu-aeg
TCP	00:08,3	01:40,2	02:08,3
HTTP	00:09,5	01:43,4	02:09,5
UDP	00:08,2	01:35,0	02:08,2

Joonis 19. 2-minutilise klipi vaatamise aeg vs protokoll

Kõige pikem aeg (Joonis 19) alates kasutaja hiireklõpsust RealPlayeris nupule PLAY kuni klipi ettemängimise alustamiseni oli HTTP korral. TCP ja UDP olid võrdsed. UDP sai ka kõige kiiremini vajalikud paketid kohale toimetatud. Kogu-aeg oli võrdne.

Pakettide arv

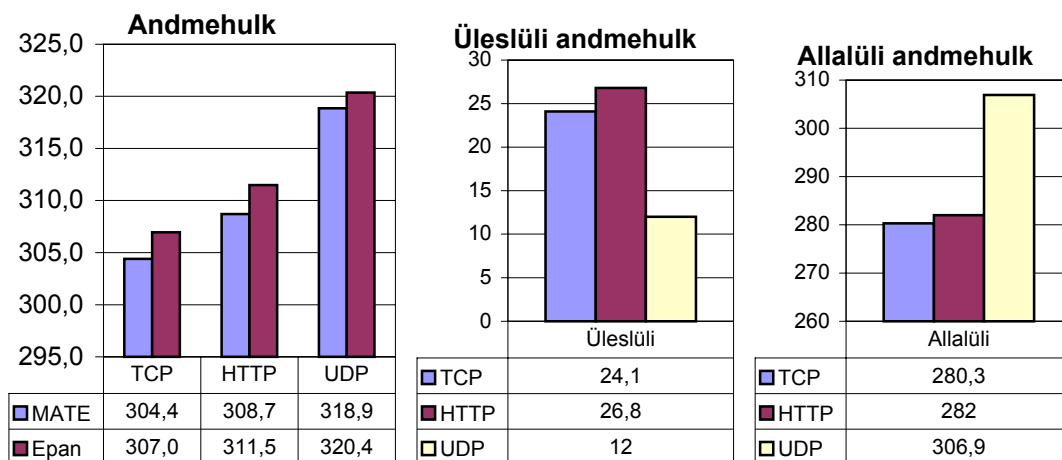


Joonis 20. Pakettide arv vs protokoll. Pakettide arv üleslülis vs protokoll. Pakettide arv allalülis vs protokoll.

RealPlayer soovis TCP ja HTTP korral kätte saada 30 võrra vähem pakette, kui UDP korral. Kogu sessiooni jooksul loendati MATE ja Epani poolt kõige vähem pakette TCP korral. UDP korral oli sessioonis palju rohkem RealPlayeriga mitteseotud pakette. Üleslülil ja allalülil pakette eraldi loendas MATE.

Andmehulk.

Andmehulka mõõdeti kilobaitides (kB). Üleslülil ja allalülil andmehulka eraldi mõõtis MATE.

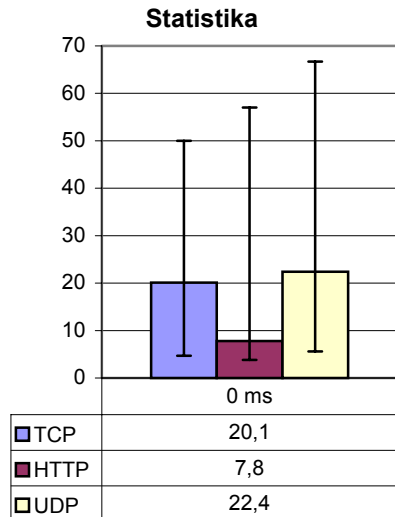


Joonis 21. Edastatud andmehulk.

TCP korral oli edastatud andmehulk natuke väiksem kui HTTP korral. UDP korral oli edastatud andmehulk 10 kB võrra suurem kui HTTP korral ning 14 kB võrra suurem kui TCP korral. Kuid UDP korral RealPlayer vajab 33 andmepaketti rohkem teistest, millest ka suurem andmehulk.

Edastuskiiruse statistika

Edastuskiirust mõõtis RealPlayer ühikutes kilobitti sekundis (kb/s).



Joonis 22. RealPlayeri edastuskiiruse statistika

Kõige suuremad edastuskiirused saavutas UDP. HTTP korral tuli kõige väiksem keskmine edastuskiirus.

2.3.2 Ettemängimise kvaliteet

Siin hinnatakse klipi ettemängimise kvaliteeti selle järgi, kas tekkis ettemängimisel katkestusi. Katkestused tekivad ettemängimise puhvri tühjenemise tõttu, mis tekib võrgupiirangute tõttu andmete saabumisest ettemängimiseks mittepiisaval hulgal.

Latentsus: 500 ms, 1500 ms, 3000 ms; üleslüli latentsus: 1500 ms, 3000 ms; allalüli latentsus: 1500 ms, 3000 ms.

Latentsuste 500 ms ja 1500 ms korral ei tekkinud ettemängimisel katkestusi. TCP ja HTTP kasutamisel tekkisid ettemängimisel katkestused, kui latentsus oli 3000 ms, UDP kasutamise korral ei tekkinud. Testitud üleslüli latentsuse korral ei tekkinud ettemängimise katkestusi TCP ja UDP korral. Kui kasutati HTTP protokoll, siis mõnel katsel latentsuse 3000 ms juures tekkis ettemängimise katkestus. Allalüli latentsuse korral ei tekkinud 2-minutilise klipi ettemängimise katkestusi.

Läbilaskevõime: 40 kb/s, 20 kb/s, 12 kb/s; üleslüli läbilaskevõime: 10 kb/s, 3 kb/s, 1 kb/s; allalüli läbilaskevõime: 20 kb/s, 12 kb/s, 7 kb/s

Läbilaskevõime 20 kb/s tekkis ettemängimisel katkestusi TCP ja HTTP transpordiprotokollide korral ja väiksema läbilaskevõime korral kõikide transpordiprotokollidega. Üleslüli läbilaskevõime piiramise korral klipi ettemängimisel katkestusi ei tekkinud ühegi transpordiprotokoll korral, küll aga pikenes aeg hiireklikist pildi nägemiseni kuni rohkem kui poole minutini üleslüli läbilaskevõimel 1 kb/s. Allalüli läbilaskevõime piiramisel 12 kb/s esines ettemängimise katkestusi kõikide transpordiprotokollide korral. Ka allalüli läbilaskevõime 7 kb/s korral on kogu 2-minutiline klipp võimalik ära vaadata 4-5 minuti jooksul koos ettemängimise katkestustega.

Paketikadu: 1%, 10%; üleslüli paketikadu: 1%, 10%, 30%; allalüli paketikadu: 1%, 10%, 30%.

Paketikao 10% korral tekkis ettemängimise katkestus HTTP korral 1 katsel ja 2 katsel ei tekkinud. Üleslüli paketikao 10% korral tekkis ettemängimise katkestus TCP ja HTTP korral samuti 1 katsel ja 2 katsel ei tekkinud. Paketikao 30% korral TCP ja

HTTP korral pooled katsed jäid pooleli, sest RealPlayer sai ajalõpu, ja pooltel katsetel õnnestus 2-minutiline klipp mõnede ettemängimise katkestustega lõpuni vaadata. UDP korral paketikao 30% korral 5 katse jooksul RealPlayer sai ajalõpu. Allalüli paketikao 1% ja 10% korral ettemängimise katkestusi ei esinenud ühegi transpordiprotokolliga kasutamisel. Allalüli paketikao 30% korral ajalõpp saadi umbes kolmandikul kordadel enne ettemängimise alustamist. Kui ettemängimine algas, siis väikeste katkestustega sai 2-minutilise testklipi ära vaadata 2-7 minuti jooksul.

Bitivea tõenäosus: 1e-9, 1e-6, 1e-5; üleslüli bitivea tõenäosus: 1e-6, 1e-5; allalüli bitivea tõenäosus: 1e-5.

Testitud bitivea tõenäosuste korral klipi ettemängimisel katkestusi ei tekkinud ühegi transpordi protokolliga korral. Samuti ei tekkinud ettemängimise katkestusi üleslüli ja allalüli bitivea tõenäosuste korral. Kui aga bitivea tõenäosus suurendada 10^{-4} , siis RealPlayer ei suuda serveriga ühendustki luua. Ühenduse loomiseks kasutatakse TCP protokolliga ja see ei tööta bitivea tõenäosuse 10^{-4} ja suurema korral.

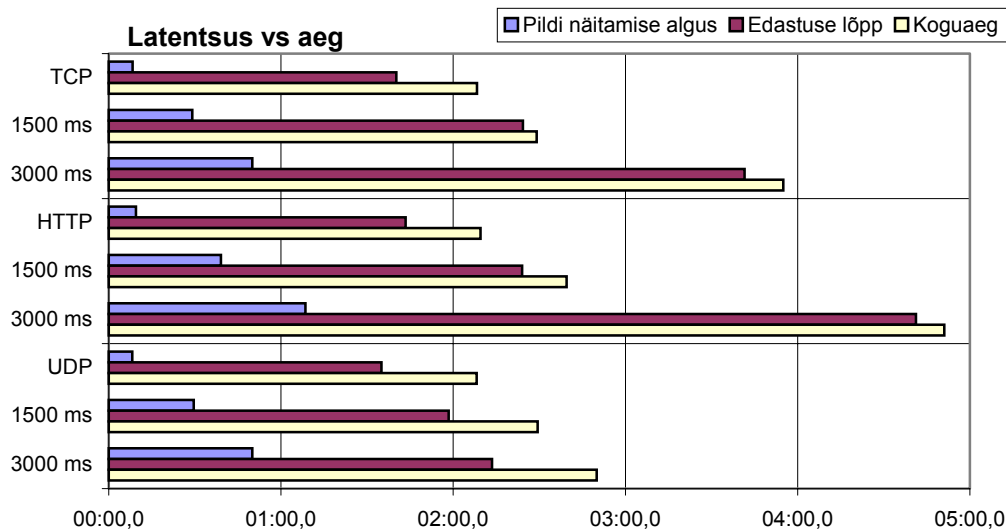
GPRS: normGPRS (latentsus 500 ms, üleslüli läbilaskevõime 20 kb/s, allalüli läbilaskevõime 40 kb/s, muud piiramata), badGPRS (latentsus 750 ms, üleslüli läbilaskevõime 10 kb/s, allalüli läbilaskevõime 20 kb/s, paketikadu 1%, bitivea tõenäosus 1e-5), bench4 (latentsus 2000 ms, üleslüli läbilaskevõime 5 kb/s, allalüli läbilaskevõime 12 kb/s, paketikadu 10%, bitivea tõenäosus 1e-5).

Võrgu parameetritega normGPRS ettemängimise katkestusi ei tekkinud ühegi transpordiprotokolliga. Võrgu parameetritega badGPRS tekkis ettemängimise katkestusi TCP ja HTTP korral ning UDP korral ettemängimise katkestusi ei tekkinud. Võrgu parameetrite bench4 korral tehti iga transpordiprotokolliga 5 katset ja HTTP korral 2-minutilist klippi lõpuni vaadata ei saanudki. TCP korral oli 2 edukat katset kestusega 12 minutit ja 16 minutit. UDP korral oli 3 edukat katset kestusega alla 3 ja poole minuti. Kõikide transpordiprotokollide korral esines ettemängimise katkestusi.

2.3.3 Aeg

Kuidas võrguparameetrid mõjutavad klipi vaatamiseks kulutatud aja pikkust. Graafikute horisontaalteljel on aeg. Vertikaalteljel on protokoll nimi (TCP, HTTP või UDP) ja tulemused piiramata võrguga ning selle all sama protokoll ja võrguparameetri väärtus. Graafikul on kolm aega: aeg kasutaja hiireklõpsust RealPlayeri nupul PLAY kuni ettemängimise alustamiseni (tähistatud kui Pildi näitamise algus), klipi andmevoo edastuse lõpp (Edastuse lõpp) ja ettemängimise lõpp (Koguaeg).

2.3.3.1 Latentsus

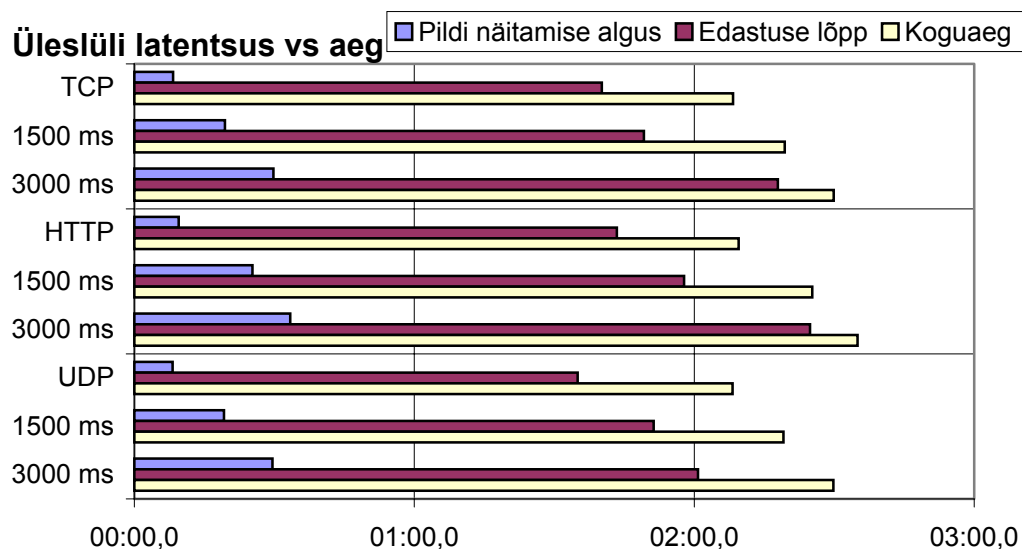


Joonis 23. Latentsus vs aeg

Aeg kasutaja hiireklõpsust kuni pildi nägemiseni on kõigi transpordiprotokollide korral suurenenud. HTTP korral on see aeg kõige suurem. TCP ja UDP korral on see aeg ligikaudu võrdne.

Klippi andmete edastuse lõpu aeg pikeneb samuti, kuid UDP korral vähem kui TCP ja HTTP korral. Kuna latentsuse 3000 ms korral oli TCP ja HTTP korral ettemängimises katkestusi, siis nendel kordadel ajad on palju pikemad. Näha on, et latentsuse 1500 ms korral TCP korral on väga väike erinevus edastuse lõpu ja koguaaja vahel, st juba suurem latentsus võib tekitada ettemängimise katkestusi, kuna andmeid ei saabu piisava kiirusega. UDP korral oli andmete saabumise kiirus piisav katkestuseta ettemängimiseks.

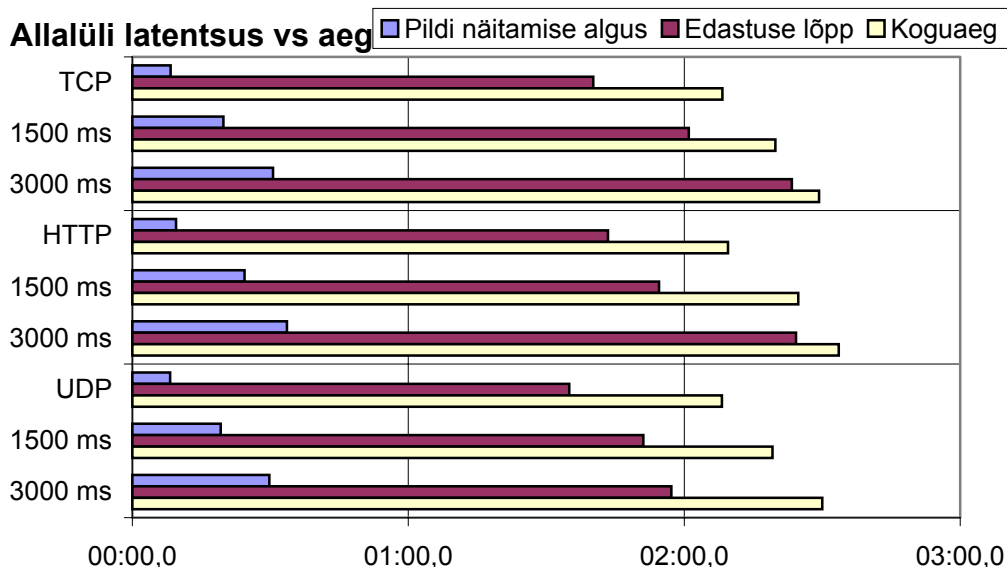
Koguaeg UDP korral kõige lühem ja kõige vähem mõjutatud latentsuse poolt.



Joonis 24. Üleslüli latentsus vs aeg

Üleslüli latentsus on aega kasutaja hiireklõpsust kuni ettemängimise alguseni mõjutanud poole vähem. Edastuse lõpu aeg on TCP ja HTTP korral samuti palju

rohkem suurenenud üleslüli latentsuse edasisel suurendamisel. Koguaaja suurenemine tekib kõikidel juhtudel pildi näitamise alguse aja suurenemisest.



Joonis 25. Allalüli latentsus vs aeg

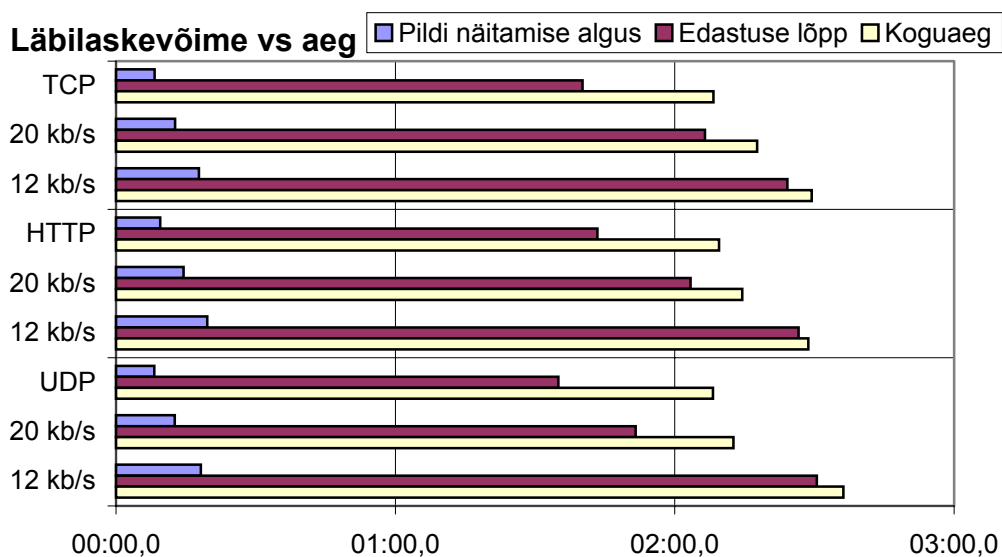
Pildi näitamise alguse ajad on sarnased üleslüli latentsuse korral saadud aegadega. Edastuse lõpu aeg on allalüli latentsuse 3000 ms TCP korral väga lähedane koguaajale. Koguajad on tulnud HTTP korral veidi suuremad, kui teistel transpordiprotokollide korral.

Latentsus – kokkuvõte

Üleslüli ja allalüli latentsus eraldi mõjutavad pildi näitamise aega ja koguaega samapalju, kui poole väiksem kogulatentsus (mõlemale suunale rakendatud). Edastuse lõpu aegadest mõjutab latentsus kõige vähem UDP korral.

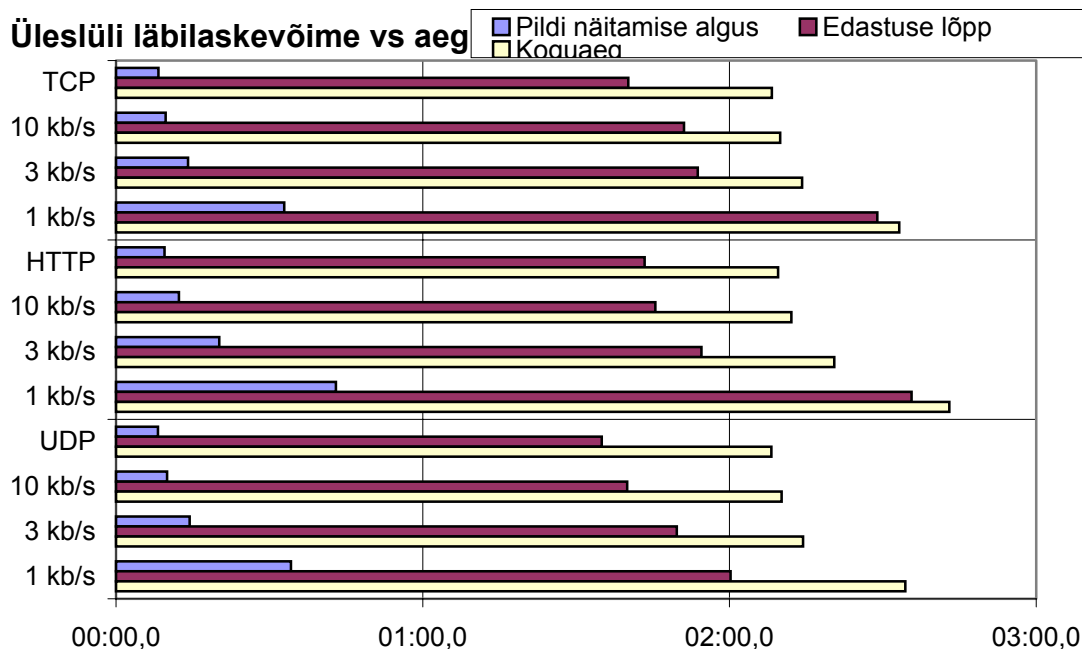
Latentsus pikendab paketi edastamiseks kuluvat aega. TCP ja HTTP korral tuleb aeg pikem nende protokollide tööalgoritmi tõttu: uusi pakette ei saa enne välja saata, kui eelmiste kohta pole kinnitust saabunud.

2.3.3.2 Läbilaskevõime



Joonis 26. Läbilaskevõime vs aeg

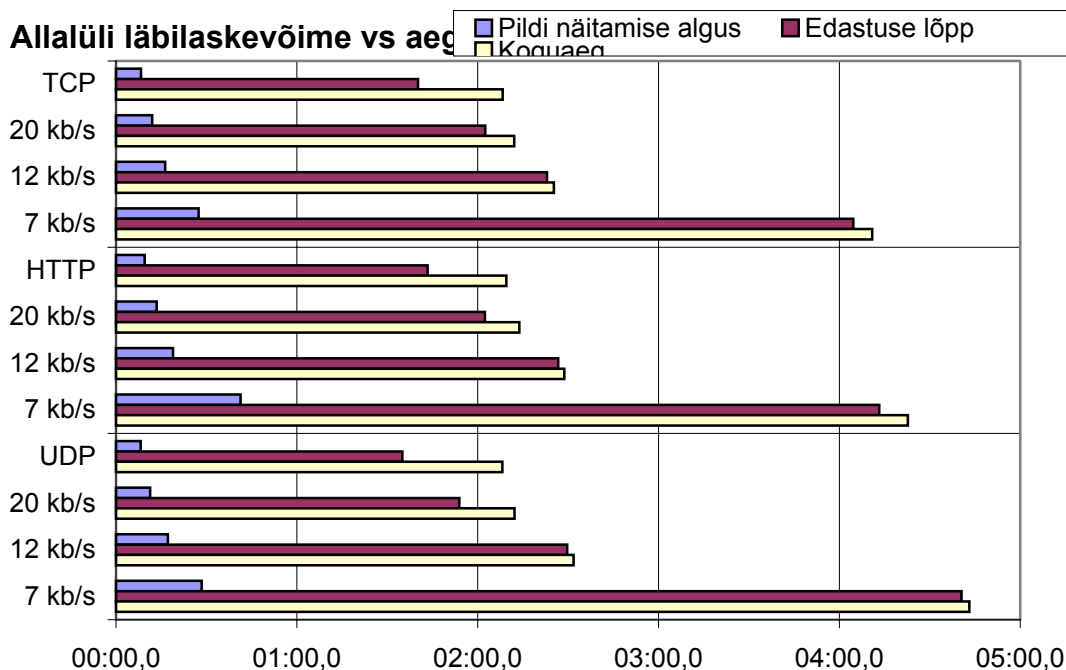
Pildi näitamise algus on kõikide transpordiprotokollide korral ühtemoodi mõjutatud läbilaskevõimest. Edastuse lõpu aeg on suurenenud kuni läbilaskevõime 12 kb/s korral on ka UDP edastuse lõpu aeg teistega ligikaudu võrdses saanud, kui läbilaskevõime 20 kb/s korral oli UDP edastuse lõpu aeg teistest väiksem ja siis ettemängimise katkestusi ei tekkinud. Kogu aeg ja edastuse lõpu aeg saavad ligilähedasteks läbilaskevõime 12 kb/s ja kõigi protokollide korral.



Joonis 27. Üleslüli läbilaskevõime vs aeg

Pildi näitamise alguse aeg on väga suureks läinud nii väikese kui 1 kb/s üleslüli läbilaskevõime korral, kuid ettemängimisel katkestusi ei tekkinud. Pildi näitamise alguse aeg on kõikide transpordiprotokollide korral ühtemoodi pikenenud ja HTTP korral teistest natuke pikem. Edastuse lõpu aeg on UDP korral vähem pikenenud kui teiste korral, kui saanud lähedaseks koguaajale. Kogu aeg on HTTP korral kõige pikem

ja erinevus teiste protokollide koguajaga suureneb üleslüli läbilaskevõime edasisel piiramisel.



Joonis 28. Allalüli läbilaskevõime vs aeg

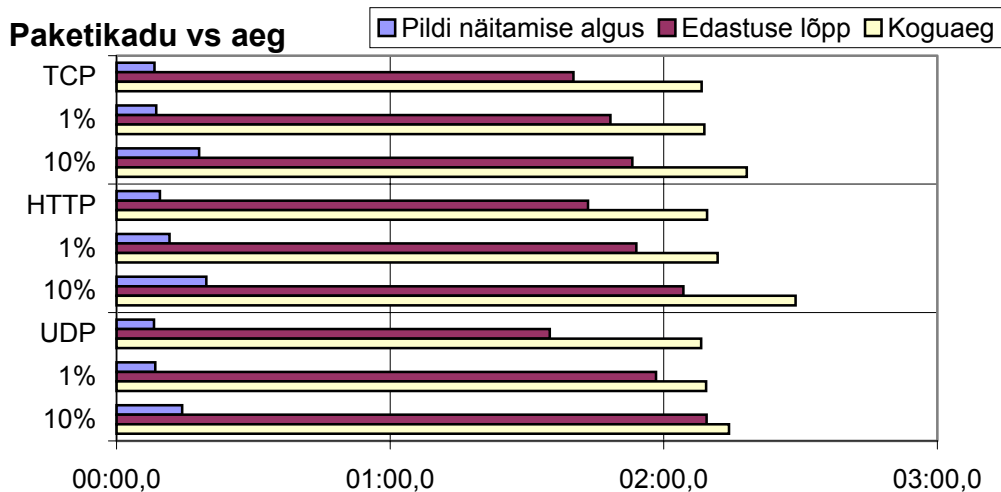
Pildi näitamise alguse aeg pikeneb märgatavalt allalüli läbilaskevõime 7 kb/s ja HTTP korral. Edastuse lõpu aeg on kõige rohkem allalüli läbilaskevõimest mõjutatud UDP korral. Alates allalüli läbilaskevõimest 12 kb/s on kõikide protokollide korral edastuse lõpu ajad ligilähedased koguaegadele. UDP korral on koguaeg allalüli läbilaskevõime 7 kb/s juures teistest pikemaks saanud.

Läbilaskevõime – kokkuvõte

Läbilaskevõime piiramine mõjutab aegu palju. Läbilaskevõime 20 kb/s võib olla juba kriitiline ettemängimise katkestuse tekkimiseks. Üleslüli läbilaskevõimet võib muuta päris väikeseks, kui samal ajal allalüli läbilaskevõime on piiramata. Kui üleslüli läbilaskevõime piiramine mõjutab rohkem TCP ja HTTP aegu, siis allalüli läbilaskevõime suur piiramine mõjutab UDP aegu rohkem kui teiste aegu.

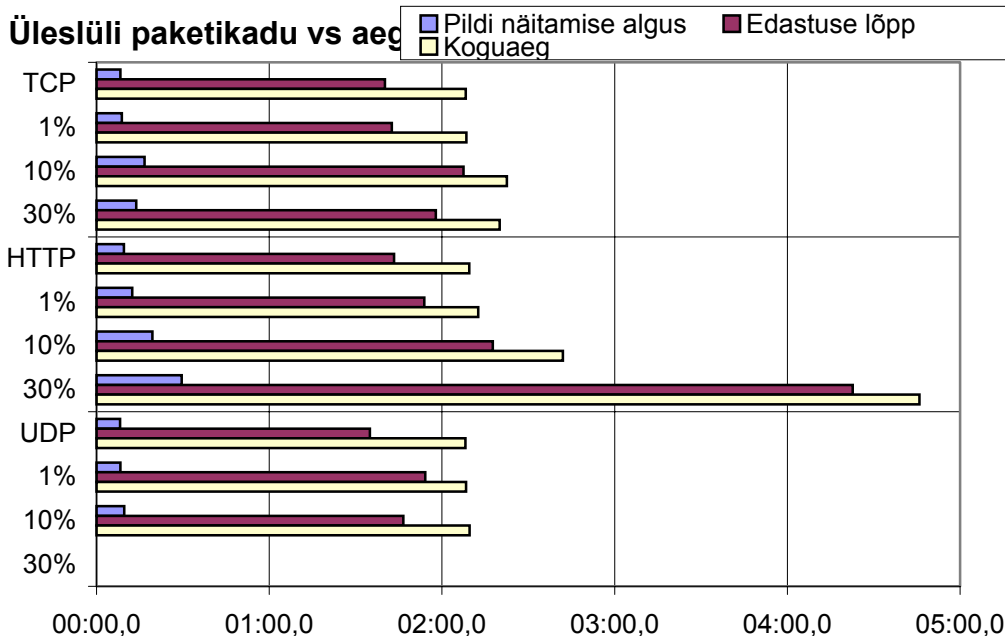
Selle põhjuseks on transpordiprotokollide töö algoritm, mille kohaselt pakette edastatakse, kui kanal on vaba. Kui kanal vaba ei ole, siis paketid jäävad ootama. Kui väga palju pakette jäävad ootama, siis ootepuhver võib täis saada (näiteks võrgu marsruuteri oma) ja need üle jäänud paketid kaovad (kustutatakse). Allalüli korral oli suurem mõju, sest UDP protokolliga ei toimu pakettide kinnituste ootamist ja seetõttu ei toimu andmete saatmise hulga reguleerimist vastavalt kinnituste saabumisele. TCP (ja HTTP) korral uusi pakette ei saadeta, kui pole eelmiste kohta kinnitusi tagasi saanud. Üleslülis ei vajata nii palju pakette, et ummistus tekiks.

2.3.3.3 Paketikadu



Joonis 29. Paketikadu vs aeg

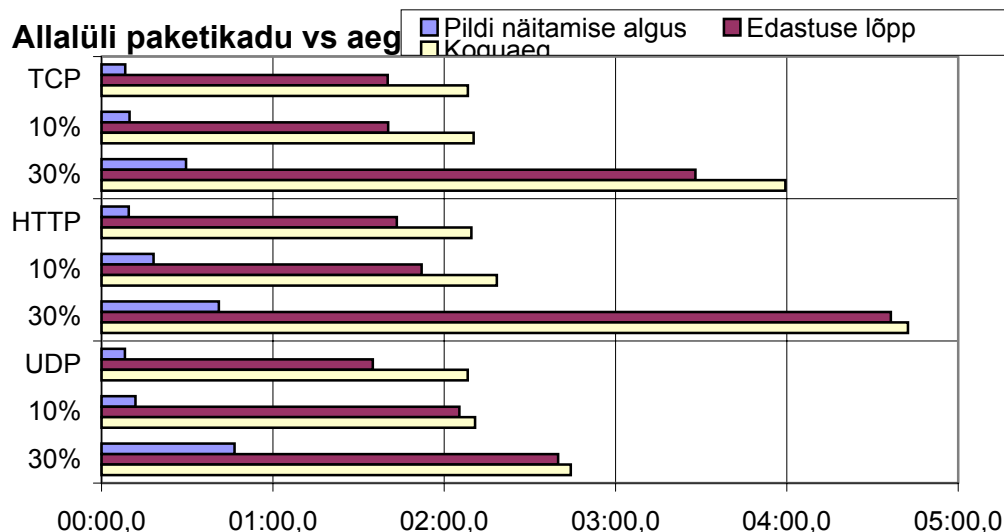
Pildi näitamise alguse aega väike paketikadu ei mõjuta, suur paketikadu mõjutab vähe. Paketikadu mõjutab edastuse lõpu aega TCP korral kõige vähem, HTTP korral natuke rohkem ja UDP korral pikeneb edastuse lõpu aeg väikse paketikao korral rohkem ja suure paketikao korral aja pikenemine aeglustub. Suure paketikao ja UDP korral edastuse lõpu ja koguaeg saavad lähedasteks. Koguaeg väikse paketikao poolt mõjutatud ei saa, suure paketikadu mõjutab HTTP korral koguaega kõige rohkem.



Joonis 30. Üleslülili paketikadu vs aeg

Üleslülili paketikadu pole UDP korral pildi näitamise alguse aega üldse mõjutanud. Väike paketikadu pole ka TCP korral pildi näitamise alguse aega mõjutanud, HTTP korral on mõju väike. Keskmine üleslülili paketikadu on TCP ja HTTP korral pildi näitamise alguse aega mõjutanud rohkem. Edastuse lõpu aeg pikeneb üleslülili paketikao suurenemisel ja kõige rohkem HTTP korral. UDP korral edastuse lõpu aeg tuli väikse paketikao korral pikem kui keskmise paketikao korral, sest RealPlayer valis väikse üleslülili paketikao korral andmevoo kiiruseks 292,4 kb/s kuni lõpuni ja

suure paketikao korral 20 kb/s, mis sai kiiremini edastatud. UDP korral koguaega üleslüli paketikadu ei mõjuta, teiste korral mõjutab vähe väikese üleslüli paketikao korral ja keskmiselt keskmise üleslüli paketikao korral. Suure üleslüli paketikao korral on ainult HTTP juures ajad väga palju pikenenud, aga UDP korral tulemust ei õnnestunud saada 5 katse korral – klippi ära vaadata ei õnnestunud.



Joonis 31. Allalüli paketikadu vs aeg

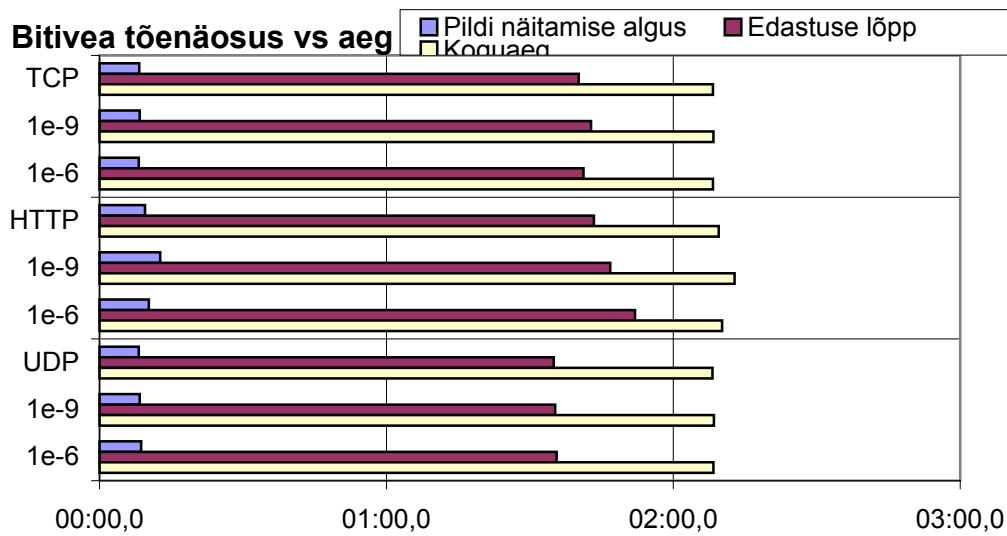
Pildi näitamise alguse aega keskmine allalüli paketikadu pole mõjutanud TCP korral, UDP korral oli mõju aja pikenedmisele väike ja HTTP korral keskmine. Suur allalüli paketikadu on pildi näitamise alguse aega kõige rohkem mõjutanud UDP korral ja teiste korral vähem. Edastuse lõpu aega on TCP ja HTTP korral keskmine paketikadu mõjutanud vähe ja UDP korral on saavutatud piir andmete saabumise ja ettemängimise juures. Suur paketikadu (30%) on kõige suuremat mõju avaldanud edastuse lõpu ajale HTTP ja TCP korral, UDP korral pole mõju nii suur. Koguaega pole keskmine allalüli paketikadu palju mõjutanud, aga suure paketikao korral on mõju suur.

Paketikadu – kokkuvõte

Väike paketikadu (1%) kas ühel suunal või mõlemal suunal ei mõjuta pildi näitamise alguse ja kogu aega, kuid UDP korral on edastuse lõpu ajale sellel kõige suurem mõju. Keskmine paketikadu mõjub kõige rohkem HTTP kasutamisel koguaegale. Suur paketikadu (30%) võimaldab nii üleslülis kui allalülis TCP ja HTTP korral lõpliku aja jooksul klipp ikkagi ära vaadata. Kuid UDP korral nii suur üleslüli paketikadu (30%) ei võimalda seda 2-minutilist klippi vaadata, aga nii suure allalüli paketikao korral saavutati ajaline tulemus kõikidest protokollidest parem.

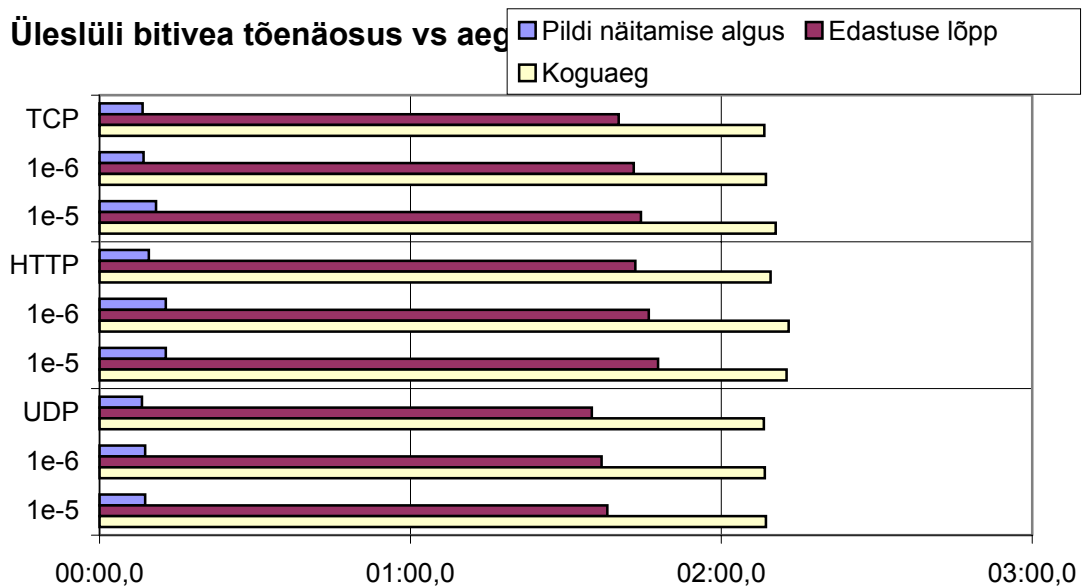
Kui pakett kohale ei saabu, siis TCP korral saatja ootab kinnitust lõpliku aja ja kuna kinnitust ei saabu, siis saadab paketi uuesti ja jääb uuesti kinnitust ootama. See kõik võtab aega. Kui RealPlayer on seatud UDP protokolliga kasutama, siis andmevoog edastatakse kasutades UDP protokolliga, aga juhtimine toimub ikka TCP protokolliga ning üleslülis liiguvad ainult TCP paketid. Paketikao 30% korral kaob umbes kolmandik pakette, mis on väga suur hulk.

2.3.3.4 Bitivea tõenäosus



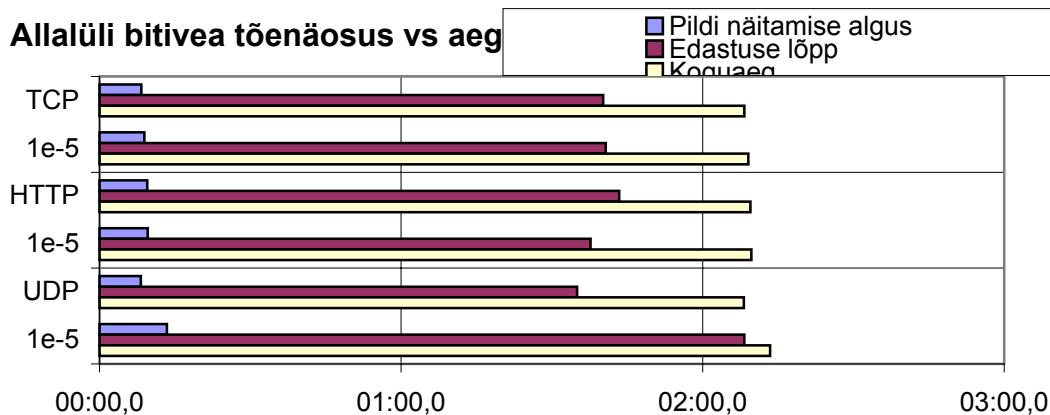
Joonis 32. Bitivea tõenäosus vs aeg

Bitivea tõenäosus pildi näitamise alguse ja koguaega ei mõjuta. Samuti bitivea tõenäosus ei mõjuta edastuse lõpu aega TCP ja UDP korral. HTTP korral bitivea tõenäosuse suurenemine toob kaasa edastuse lõpu aja pikenedmise.



Joonis 33. Üleslüli bitivea tõenäosus vs aeg

Üleslüli bitivea tõenäosuse mõju aegadele on sama, aga natuke väiksem.



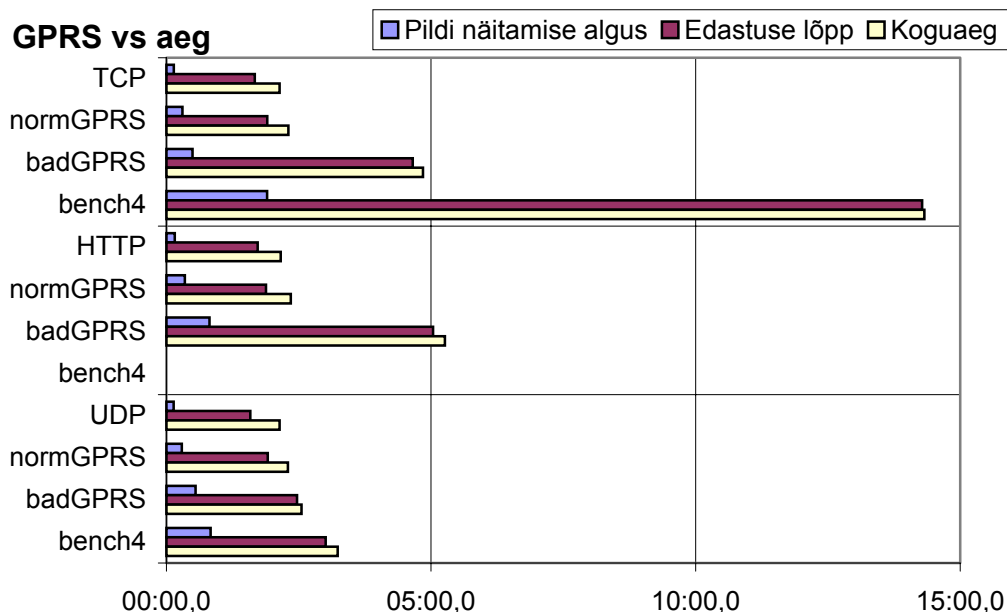
Joonis 34. Allalüli bitivea tõenäosus vs aeg

Suur allalüli bitivea tõenäosus on pikendanud edastuse lõpu aega UDP korral. Teistele transpordiprotokollidele erilist mõju ei ole.

Bitivea tõenäosus kokkuvõte

Bitivea tõenäosus aegadele palju mõju ei avalda, va bitivea tõenäosus suurem kui 10^{-4} ($1e-4$) korral, siis TCP ei loo ühendust kliendi ja serveri vahel.

2.3.3.5 GPRS



Joonis 35. GPRS vs aeg

Erinevad GPRS võrgu tingimused mõjutavad rohkem TCP ja HTTP aegu ja vähem UDP aegu.

HTTP bench4 juures pole sellepärast midagi, sest 5 katse jooksul saadi 5 korda *timeout*. Näha on, et TCP ja HTTP on kõige rohkem mõjutatud võrgust. UDP üllatavalt vähe. Veel üllatab TCP korral RealPlayeri "kannatlikkus" – ligi 15 min raisata 2-minutilise klipi ettemängimiseks.

2.3.3.6 Aeg – kokkuvõte

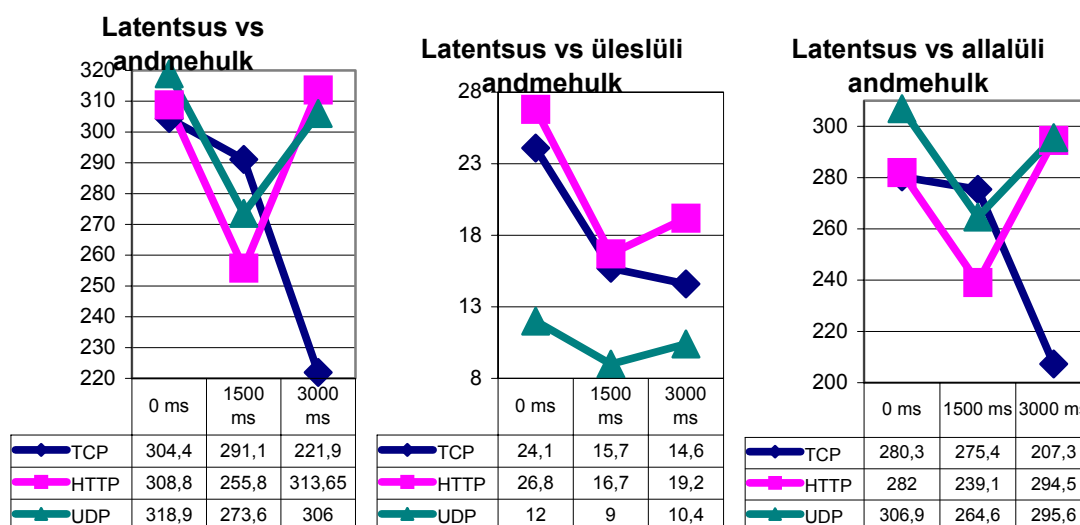
Latentsus mõjutab aega keskmiselt, TCP ja HTTP korral rohkem kui UDP korral. Võrgu läbilaskevõime mõjutab aega keskmiselt. Allalüli läbilaskevõime mõjutab aega rohkem kui üleslüli läbilaskevõime. Bitiviga ei mõjuta aega peaaegu üldse. Paketikadu mõjutab aega keskmiselt, aga kui paketikadu on suurem, kui 10%, siis klipi vaatamine võib mitte võimalik olla. Allalüli paketikadu mõjutab rohkem TCP ja HTTP kasutamise kui UDP kasutamise korral. Erinevate GPRS tingimuste korral on UDP kõige kiiremini toime tulnud.

2.3.4 Andmehulk

Andmehulk näitab, kui palju me teenuse eest maksame, kui maksustatakse edastatud andmehulga järgi. Kui maksustatakse võrgukasutamise aja järgi, siis loodame, et saaks võimalikult suur andmehulk vahetatud võimalikult lühikese aja jooksul. Inimesele, kes seda klippi vaatab, on kasulik allalüli andmehulga suurus. Mida suurem see on, seda kvaliteetsem on klipp.

Graafikutel on horisontaalteljel võrguparameetri väärtus võrgutingimuste halvenemise suunas ja vertikaalteljel on andmehulk kilobaitides (kB). Vasakpoolsel graafikul on üleslüli ja allalüli andmehulk kokku, keskmisel on üleslüli andmehulk ja parempoolsel on allalüli andmehulk.

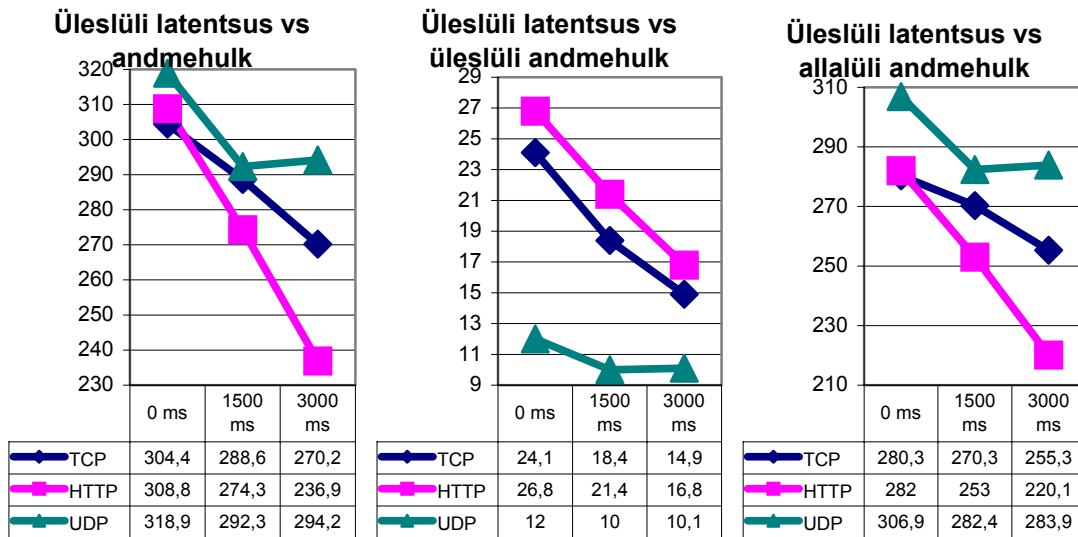
2.3.4.1 Latentsus



Joonis 36. Latentsus vs andmehulk. Latentsus vs üleslüli andmehulk. Latentsus vs allalüli andmehulk

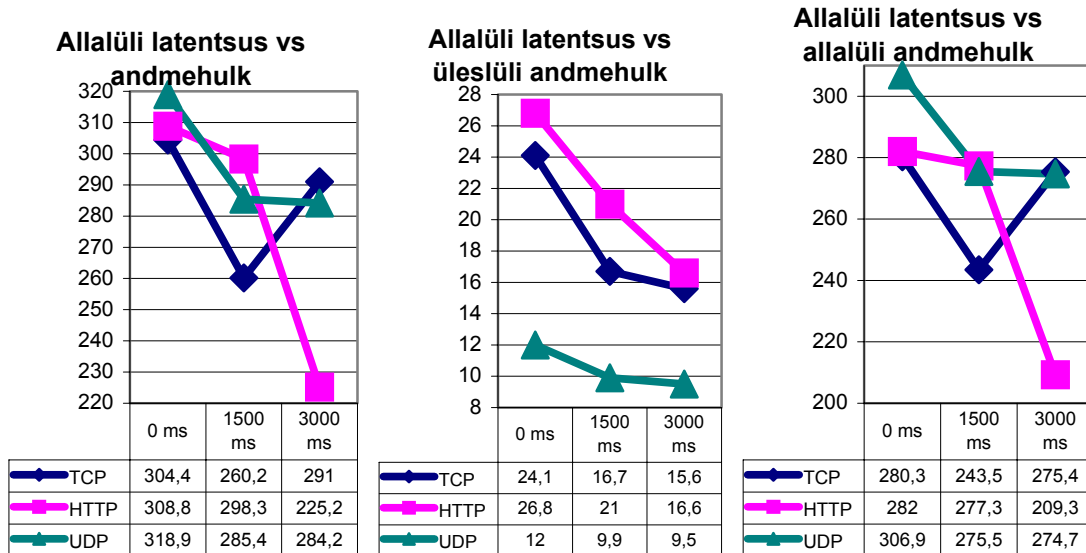
TCP korral andmehulk väheneb monotoonselt nii üleslülis (Joonis 36 keskmine graafik) kui allalülis (Joonis 36 parempoolne graafik). Teiste korral pärast 1500 ms latentsuse juures vähenemist suureneb latentsuse 3000 ms juures. Põhjus on sellest, millisel ajal ettemängimise käigus lülitati kiirusega 292,4 kb/s kodeeritud andmevoolt 20 kb/s või mõnel juhul kiirusega 15 kb/s ja 12 kb/s kodeeritud andmevoole. Latentsusel 1500 ms HTTP korral 3 katses 2 soovis lõpuks RealPlayer saada 15 kb/s andmevoogu ja 1 korral 20 kb/s. Ning latentsuse 3000 ms TCP korral RealPlayer soovis saada mõlemal 2 katsel andmevoogu kiirusega 12 kb/s. UDP juures võib põhjus olla jälle RealPlayeris, mis soovis saada piiramata võrgu korral 739

paketti, 1500 ms latentsuse korral 683 paketti ja 3000 ms latentsuse korral 719 paketti.



Joonis 37. Üleslüli latentsus vs andmehulk. Üleslüli latentsus vs üleslüli andmehulk. Üleslüli latentsus vs allalüli andmehulk.

Üleslüli latentsus on kõikide transpordiprotokollide korral andmehulka mõlemas suunas vähendanud. Kõige vähem on allalüli latentsus vähendanud üleslüli andmehulka UDP korral (Joonis 37 keskmise graafik).



Joonis 38. Allalüli latentsus vs andmehulk. Allalüli latentsus vs üleslüli andmehulk. Allalüli latentsus vs allalüli andmehulk.

Allalüli latentsus on samuti üldiselt andmehulka vähendanud. Suur allalüli latentsus pole UDP korral andmehulka edasi vähendanud. (Joonis 38)

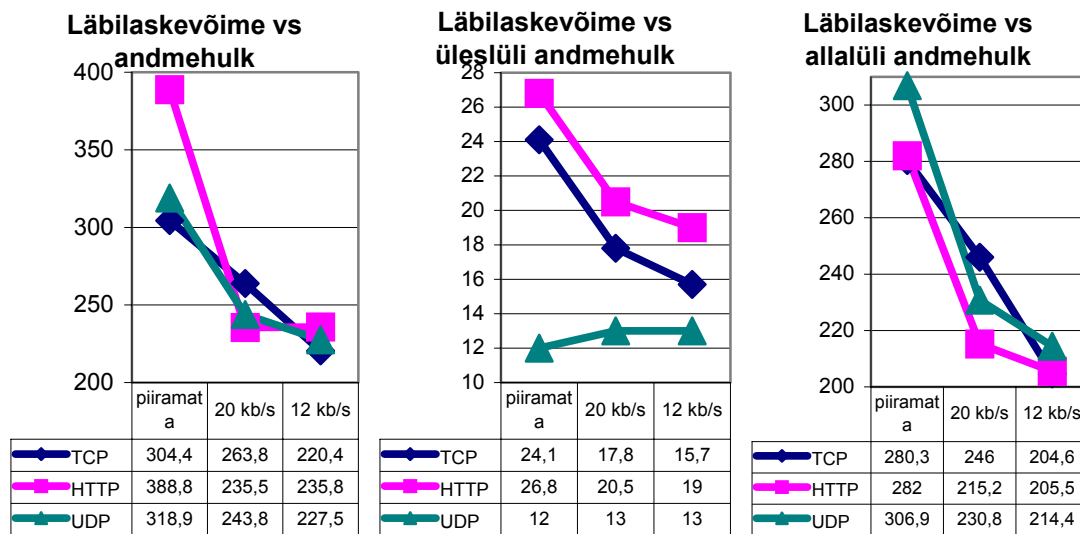
TCP korral väiksema latentsuse korral on andmehulk vähenenud ja suure latentsuse korral suurenenud esialgsele tasemele (Joonis 38 vasakpoolne graafik). Seda on mõjutanud allalüli latentsuse 1500 ms juures kodeeritud andmevoogi kiirus 15 kb/s, kui

teiste protokollide ja üleslüli latentsuse 3000 ms korral on kodeeritud andmevoo kiirus 20 kb/s.

Latentsus – kokkuvõte

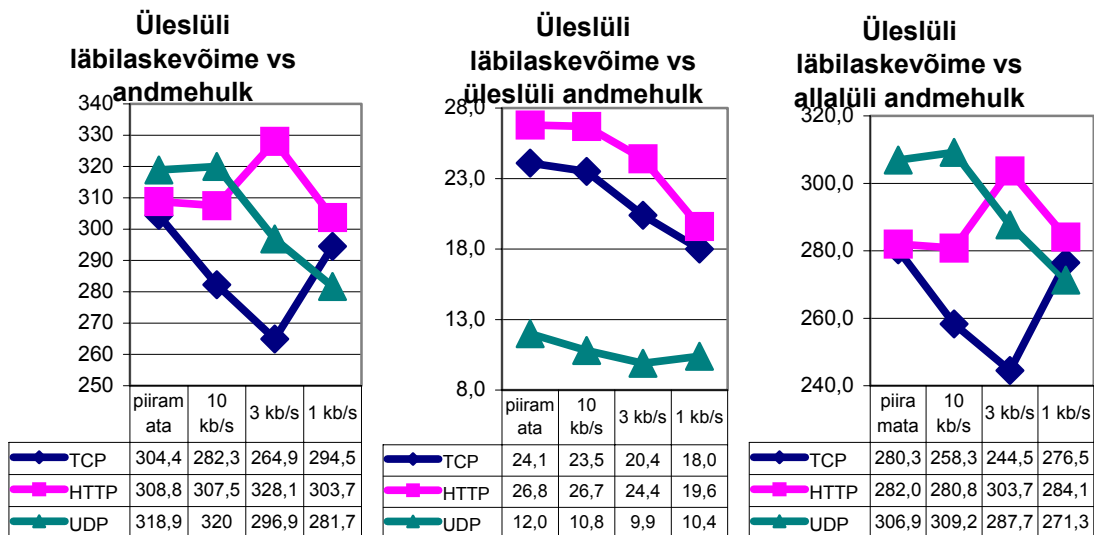
Andmehulga suurst on mõjutanud see, millise kodeeritud kiirusega andmevoogu RealPlayer vajab ja millal lülitatakse esialgselt 292,4 kb/s kiiruselt ümber väiksemale, kas 20 kb/s, 15 kb/s või 12 kb/s. See mõjutab eelkõige allalüli andmehulka. Kui esialgselt kodeeritud andmevoo kiiruselt ümber ei lülitata, siis andmehulk jääb kõige väiksemaks. Latentsus üldiselt vähendab andmehulka. Allalüli latentsus vähendab allalüli andmehulka rohkem kui üleslüli latentsus allalüli andmehulka. UDP korral latentsus mõjutab andmehulka kõige vähem ja üleslüli andmehulka ei mõjuta peaaegu üldse.

2.3.4.2 Läbilaskevõime



Joonis 39. Läbilaskevõime vs andmehulk. Läbilaskevõime vs üleslüli andmehulk. Läbilaskevõime vs allalüli andmehulk

Üldiselt läbilaskevõime vähendamisel andmehulk väheneb. UDP korral üleslüli andmehulk suureneb. Allalüli korral kõigil väheneb.

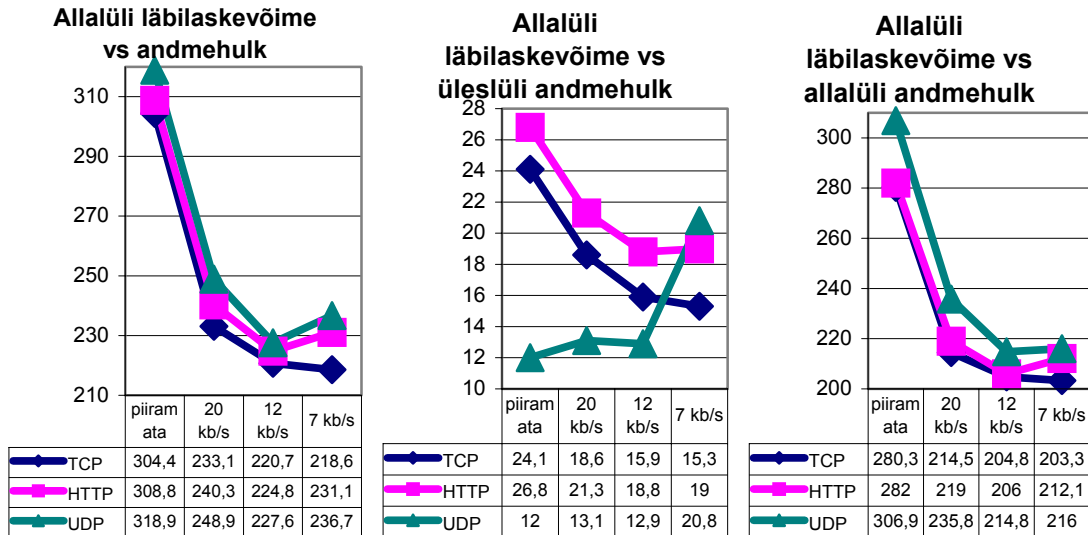


Joonis 40. Üleslüliläbilaskevõime vs andmehulk. Üleslüliläbilaskevõime vs üleslüliläbilaskevõime. Üleslüliläbilaskevõime vs allalüliläbilaskevõime.

Üleslüliläbilaskevõime mõjutab kogu andmehulka samapalju kui allalüliläbilaskevõime. Allalüliläbilaskevõime UDP korral, võib öelda, jääb samaks, TCP ja HTTP korral monotoonselt väheneb.

TCP korral jäi lõpuks soovitud kodeeritud andmevoo kiirus 15 kb/s 2 juhul 3-st, kui üleslüliläbilaskevõime oli 3 kb/s, ja 1 juhul 3-st, kui üleslüliläbilaskevõime oli 10 kb/s, ning 20 kb/s teistel üleslüliläbilaskevõimetest. Kui oleks TCP korral arvesse läinud kõikidel katsetel kodeeritud andmevoo kiirus 20 kb/s, siis allalüliläbilaskevõime jääb samaks.

HTTP korral on näha üleslüliläbilaskevõime 3 kb/s korral u 20 kB võrra suurem allalüliläbilaskevõime. Ka oli RealPlayeri statistika keskmine läbilaskevõime HTTP korral 1 katsel u 10 kb/s võrra suurem üleslüliläbilaskevõime 3 kb/s juures kui teiste üleslüliläbilaskevõimete juures. Kõikidel üleslüliläbilaskevõimetest oli ühtemoodi lõplik kodeeritud andmevoo kiirus 20 kb/s, RealPlayer statistikas soovitud pakettide arv 702-706. Kogu sessiooni pakettide arv vähenes koos üleslüliläbilaskevõime vähenemisega ja keskmine paketi pikkus suurenes. Seega võib järeldada, et katseandmed on õiged.



Joonis 41. Allalüli läbilaskevõime vs andmehulk. Allalüli läbilaskevõime vs üleslüli andmehulk. Allalüli läbilaskevõime vs allalüli andmehulk.

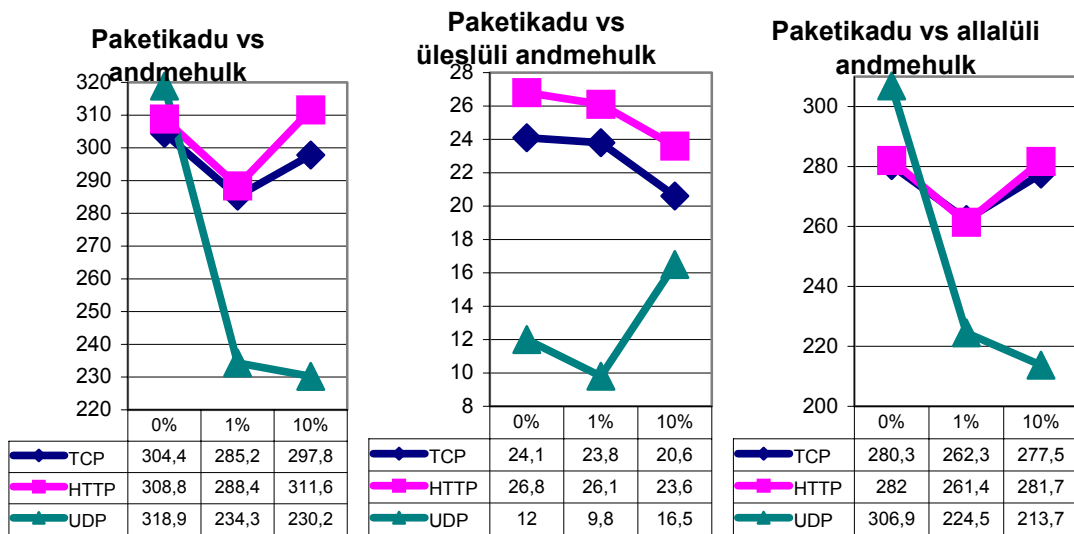
Allalüli läbilaskevõime vähendamisel kõikide protokollide korral on näha ühesugune käitumine (Joonis 41 vasakpoolne graafik). Allalüli läbilaskevõimel 12 kb/s kõikide protokollide korral on valitud kodeeritud andmevoo kiirus 15 kb/s; väiksematel läbilaskevõimel ümberlülitamist 292,4 kb/s pealt ei toimugi. Näha on allalüli läbilaskevõime 7 kb/s juures UDP üleslüli andmehulga suurenemine. Selle põhjuseks võib pidada sagedase puhverdamise jaoks vajaminevate andmete vahetust kliendi ja serveri vahel.

Läbilaskevõime – kokkuvõte

Läbilaskevõime vähenemisel andmehulk väheneb, välja arvatud juhul, kui ainult üleslüli läbilaskevõime vähendada, siis UDP korral andmehulk esialgu jääb samaks, aga siis väheneb. TCP korral andmehulk suurenes väikse üleslüli läbilaskevõime korral ja HTTP korral andmehulk oli suurem üleslüli läbilaskevõime 3 kb/s korral. Need mõlemad TCP ja HTTP andmehulgad olid mõjutatud allalüli andmehulkadest. TCP korral RealPlayer valis üleslüli läbilaskevõime 10 kb/s juures 1 korral 3-st kodeeritud kiiruseks 15 kb/s ja üleslüli läbilaskevõime 3 kb/s juures 2 korral 3-st kodeeritud kiiruseks 15 kb/s. Kõikidel teistel läbilaskevõimel ja transpordiprotokollide korral valis RealPlayer kodeeritud kiiruseks 20 kb/s. Kui arvestada ainult kodeeritud kiirust 20 kb/s, siis üleslüli läbilaskevõime 10 kb/s korral on TCP andmehulk kokku 13 kB võrra UDP omast suurem üleslüli andmehulga tõttu, ja üleslüli läbilaskevõime 3 kb/s korral on TCP andmehulk UDP ja HTTP oma vahel. HTTP korral lihtsalt oli allalüli andmehulk suurim üleslüli läbilaskevõime 3 kb/s juures kui teiste üleslüli läbilaskevõimete juures.

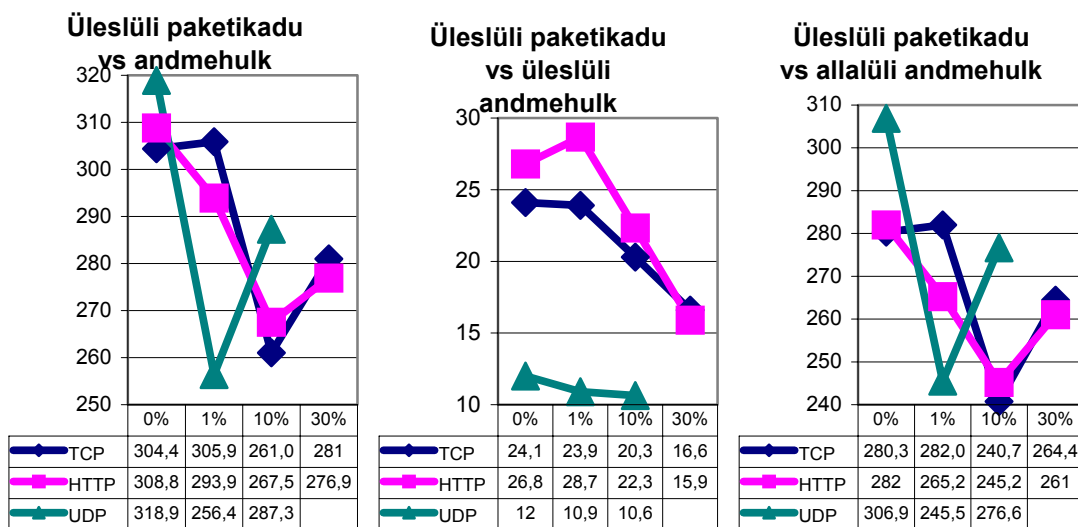
Allalüli läbilaskevõime vähenemisel UDP korral üleslüli andmehulk suureneb.

2.3.4.3 Paketikadu



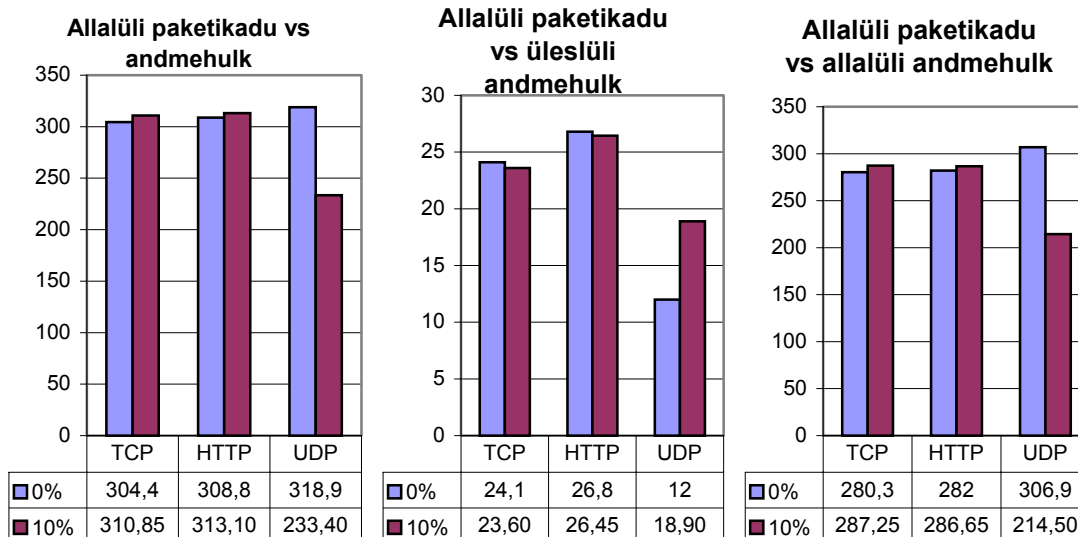
Joonis 42. Paketikadu vs andmehulk. Paketikadu vs üleslül andmehulk. Paketikadu vs allalül andmehulk.

Paketikadu esialgu andmehulka vähendab, kuid suure paketikao korral andmehulk suureneb nii TCP kui HTTP korral (Joonis 42 vasakpoolne graafik). See toimub allalül andmehulga suurenemise tõttu (Joonis 42 parempoolne graafik). Paketikao suurenedes üleslül andmehulk suureneb ainult UDP korral, allalül andmehulk väheneb ainult UDP korral.



Joonis 43. Üleslül paketikadu vs andmehulk. Üleslül paketikadu vs üleslül andmehulk. Üleslül paketikadu vs allalül andmehulk.

Paketikadu üleslülis mõjutab UDP korral ainult vähese paketikao korral, kui andmehulk väheneb, suurema üleslül paketikao korral andmehulk suureneb allalül andmehulga arvel. UDP korral üleslül paketikao 1% korral oli kodeeritud andmevoo kiirus 292,4 kb/s, teistel paketikao väärtustel 20 kb/s. Kui arvesse võtta ainult see 1 katse 3-st, kui toimus kodeeritud andmevoo kiiruse ümberlülitamine 20 kb/s peale, siis allalül andmehulk üleslül paketikao 1% korral oli 306,8 kB, st ainult natuke väiksem kui paketikao 0% korral. Üleslül paketikao 30% korral UDP korral tehti 5 katset ja klippi lõpuni vaadata ei õnnestunud.



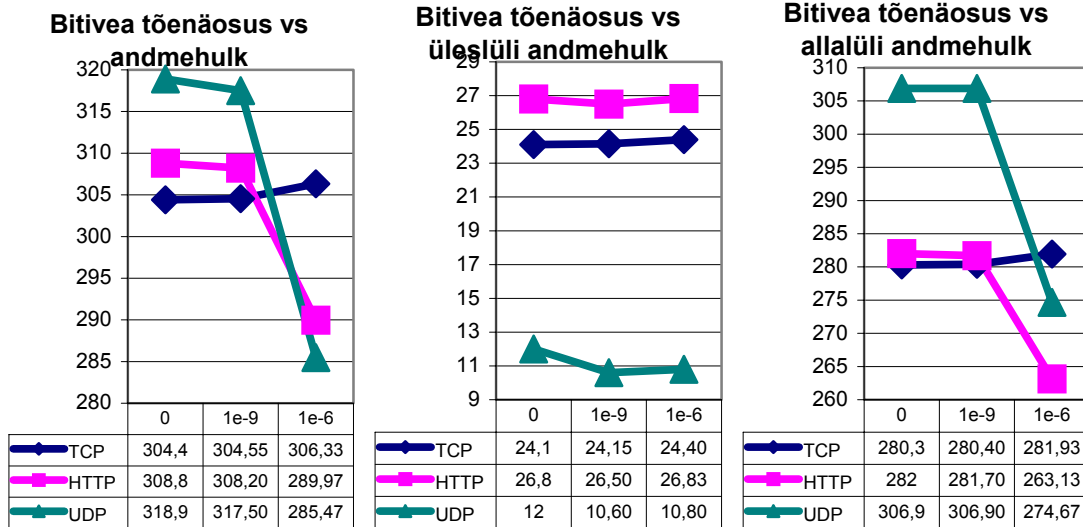
Joonis 44. Allalüli paketikadu vs andmehulk. Allalüli paketikadu vs üleslüli andmehulk. Allalüli paketikadu vs allalüli andmehulk.

Allalüli paketikadu võrreldes 0% ja 10 %, andmehulk jääb umbes samaks TCP ja HTTP korral, UDP korral aga väheneb. See vähenemine on allalüli, aga üleslüli UDP korral andmehulk suureneb paketikao suurenemisel. Selle põhjuseks on RealPlayeri poolt pakettide uuestisaatmise nõuded.

Paketikadu – kokkuvõte

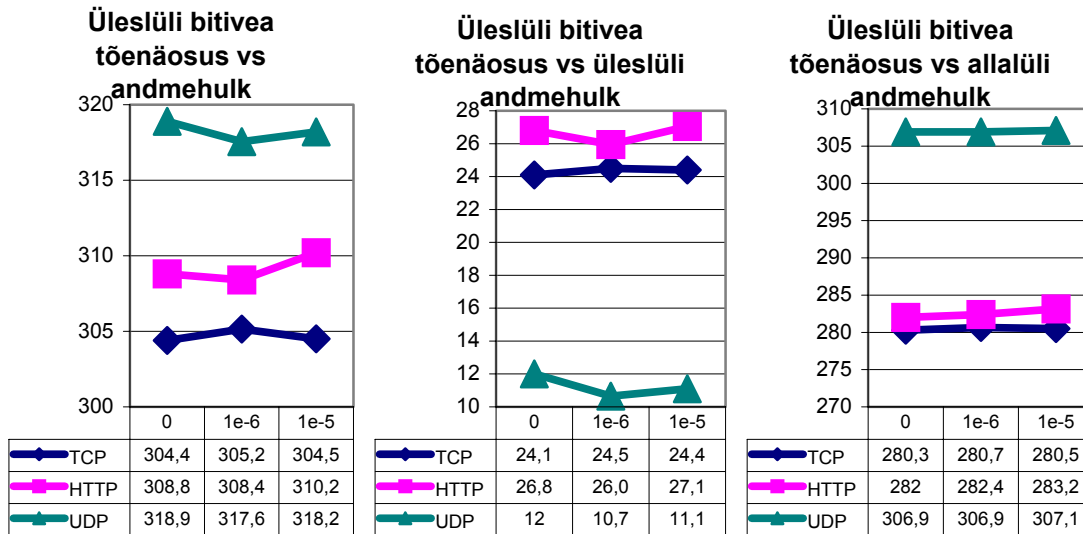
Väikese (1%) paketikao korral andmehulk väheneb, kui paketikadu suureneb kuni 10% ni, siis andmehulk suureneb taas, seda eriti UDP korral. Paketikadu üleslüli ja allalüli mõjutavad andmehulka samamoodi. Paketikadu allalüli suurendab UDP korral üleslüli andmehulka, üleslüli paketikadu UDP korral jätab üleslüli andmehulga samaks.

2.3.4.4 Bitivea tõenäosus



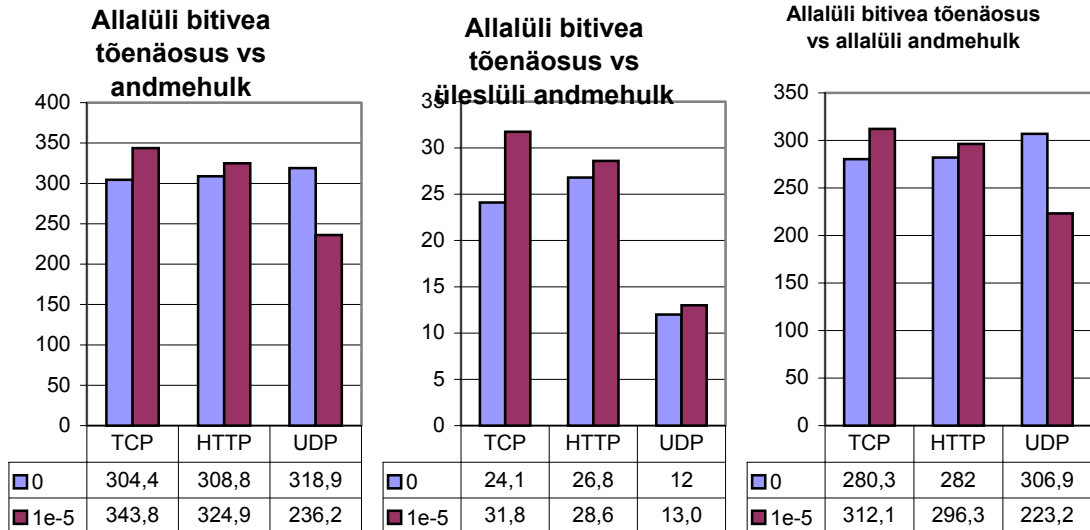
Joonis 45. Bitivea tõenäosus vs andmehulk. Bitivea tõenäosus vs üleslüli andmehulk. Bitivea tõenäosus vs allalüli andmehulk.

Bitivea tõenäosuse suurenemine üldiselt mõjutab ainult allalüli andmehulka monotoonse vähenemise suunas. Ainult TCP korral on allalüli andmehulk väga vähe suurenenud. Bitivea tõenäosus üleslüli andmehulka ei ole mõjutanud.



Joonis 46. Üleslüli bitivea tõenäosus vs andmehulk. Üleslüli bitivea tõenäosus vs üleslüli andmehulk. Üleslüli bitivea tõenäosus vs allalüli andmehulk.

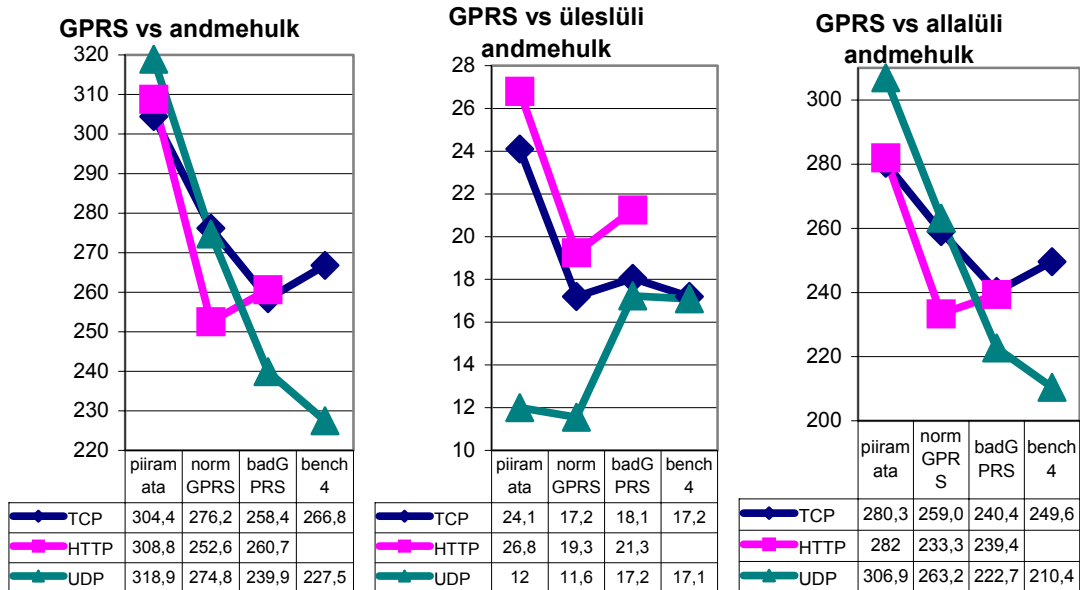
Üleslüli bitivea tõenäosuse suurenemine üldiselt andmehulka ei mõjuta. Üleslülis on väga väike andmehulga suurenemine.



Joonis 47. Allalüli bitivea tõenäosus vs andmehulk. Allalüli bitivea tõenäosus vs üleslüli andmehulk. Allalüli bitivea tõenäosus vs allalüli andmehulk.

Allalüli bitivea tõenäosus vähendab allalüli andmehulka UDP korral, teiste korral andmehulgad suurenevad.

2.3.4.5 GPRS



Joonis 48. GPRS vs andmehulk. GPRS vs üleslüli andmehulk. GPRS vs allalüli andmehulk.

GPRS võrgutingimuste halvenemise korral esialgu andmehulk väheneb, edasistel võrgutingimuste halvenemisel andmehulk suureneb.

HTTP juures bench4 tulemuse puudumine tähendab, et 5 katse jooksul ei saanud 2-minutilist testklippi täielikult ette mängida ja nii siia tulemust ei saanud. Aga üldiselt andmehulk väheneb. Kuid ka ettemängimise kvaliteet on halb, st palju katkestusi.

2.3.4.6 Andmehulk – kokkuvõte

Kui võrgutingimused halvnevad, siis üldiselt andmehulk väheneb. Latentsus mõjutab TCP korral andmehulga vähenemist kõige rohkem ja katsed näitasid, et HTTP korral latentsus andmehulka ei mõjuta. Võrgu läbilaskevõime vähendab andmehulka kõigil juhtudel. See ei suuda ületada UDP üleslüli andmehulga suurenemist (samaks jäämist). Kui läbilaskevõimet ainult üleslülis piirata, siis monotoonse andmehulga vähenemise saavutame ainult UDP korral, üleslüli andmehulga monotoonse vähenemise kõigi transpordiprotokollide korral. Selline TCP ja HTTP “eriline” käitumine tuleneb allalüli andmehulga käitumisest üleslüli läbilaskevõime piiramisel. Bitivea tõenäosus mõjutab kõige vähem HTTP korral.

Mõnikord üleslüli andmehulk suureneb ja allalüli andmehulk väheneb. Kuid oli märgata ka võrgu parameetri väikesel piiramisel andmehulga kasvu (kahanemist) ja sama parameetri suurel piiramisel andmehulga algse taseme saavutamist või kahanemist (kasvu). Sellise “anomaalia” juures tuleb vaadata teiste rakenduse parameetrite mõju uuritavale.

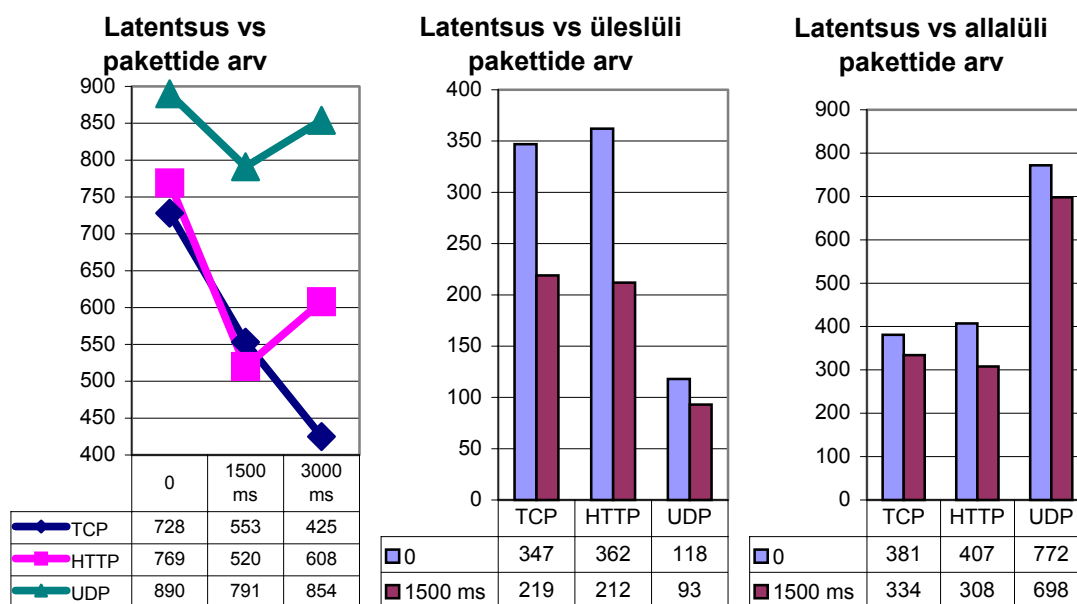
Järelkult kõige odavamalt saab läbi, kui leida sellised võrguparameetrid, kus kliipi ettemängimisel ei teki katkestusi.

2.3.5 Pakettide arv

Pakettide arv võib näidata, kui “jutukas” on rakendus. Kuna mobiilvõrkudes on suur latentsus, siis paremad on sellised rakendused, mis suudavad vähemate pakettidega öelda, mis neil vaja. Pakette loendab MATE.

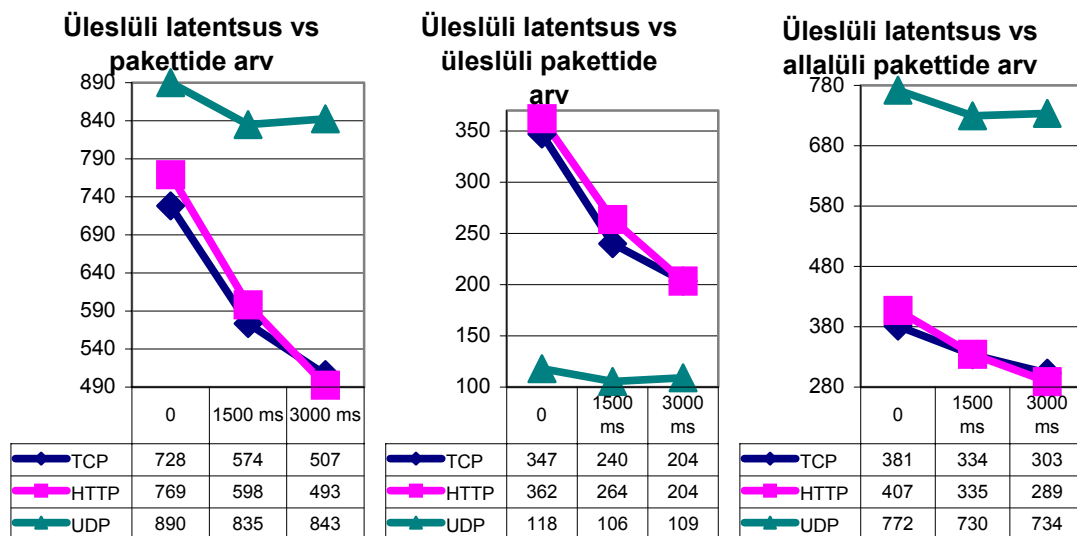
Graafikute horisontaalteljel on võrguparameetri väärtus võrgutingimuste halvenemise suunas ja vertikaalteljel on pakettide arv (ühikuta). Vasakpoolsel graafikul on üleslüli ja allalüli pakettide arv kokku, keskmisel graafikul on üleslüli pakettide arv ja parempoolsel graafikul on allalüli pakettide arv.

2.3.5.1 Latentsus



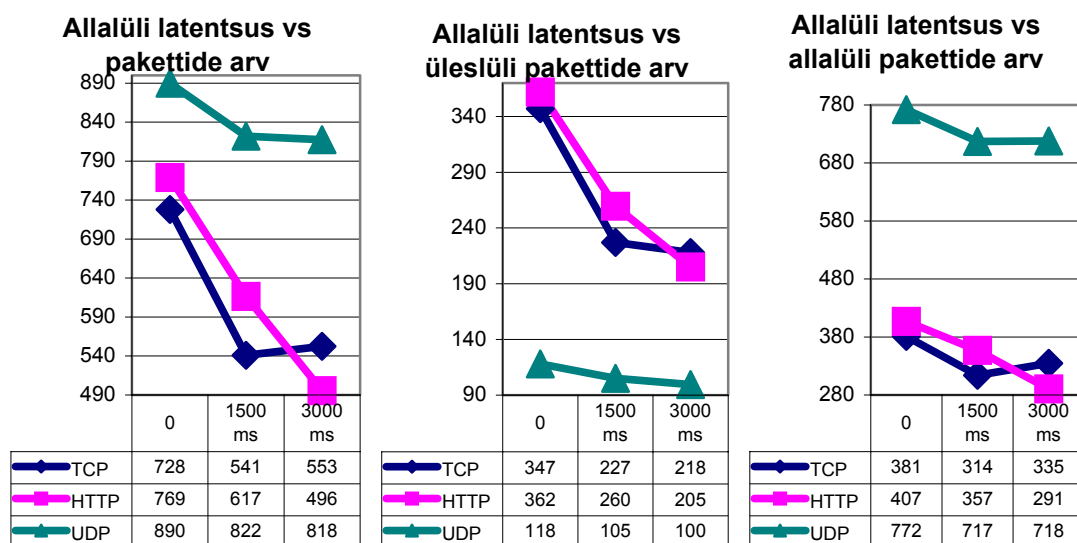
Joonis 49. Latentsus vs pakettide arv. Latentsus vs üleslüli pakettide arv. Latentsus vs allalüli pakettide arv.

Latentsuse suurenedes pakettide arv väheneb iga transpordiprotokollil korral. Latentsus on üleslülil pakettide arvu vähendanud TCP ja HTTP korral rohkem kui UDP korral, allalülil pakettide arvu natuke iga protokolli korral.



Joonis 50. Üleslülil latentsus vs pakettide arv. Üleslülil latentsus vs üleslülil pakettide arv. Üleslülil latentsus vs allalülil pakettide arv.

Üleslülil latentsus on TCP ja HTTP korral pakettide arvu vähendanud nii üleslülil kui ka allalülil. UDP korral pole mõju suur.



Joonis 51. Allalülil latentsus vs pakettide arv. Allalülil latentsus vs pakettide arv. Allalülil latentsus vs allalülil pakettide arv.

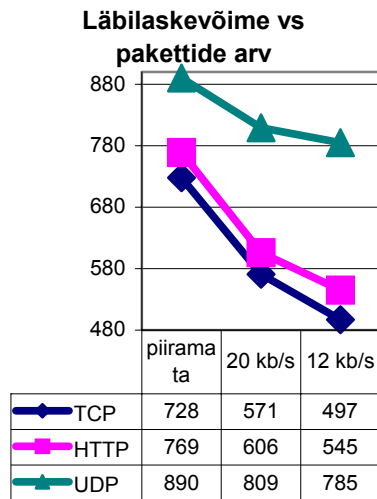
Allalülil latentsus on mõjutanud HTTP pakettide arvu kõige rohkem, teisi vähem.

Latentsus – kokkuvõte

Latentsuse suurenedes pakettide arv väheneb iga protokolli korral nii üleslülil kui allalülil. UDP korral on see vähenemine natuke väiksem.

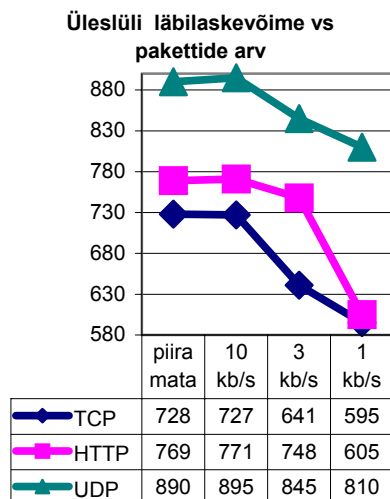
2.3.5.2 Läbilaskevõime

Teostatud testides ei olnud loetud läbilaskevõime ja üleslüli läbilaskevõime korral pakettide arvu eraldi suundadel (üleslülis ja allalülis), seetõttu neid andmeid siin pole.



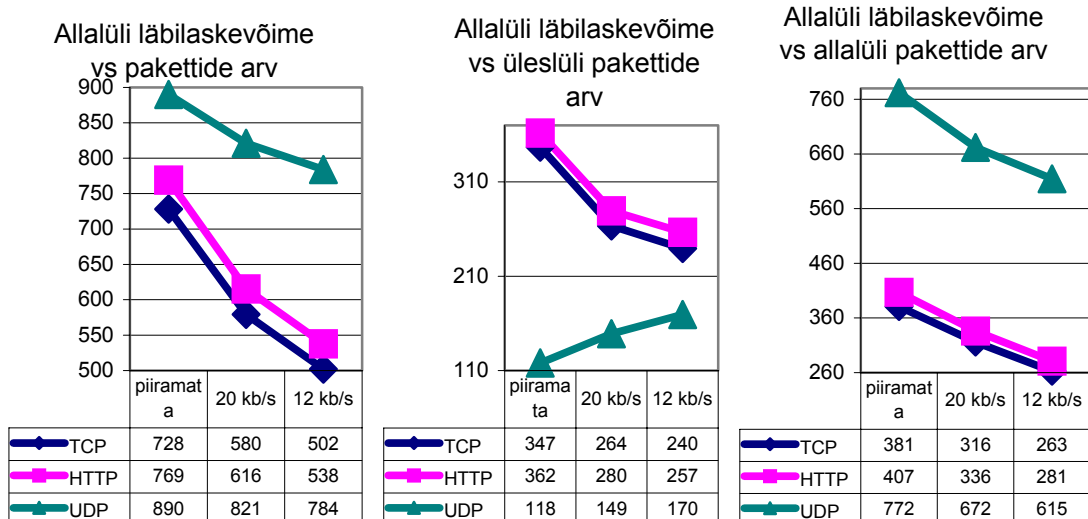
Joonis 52. Läbilaskevõime vs pakettide arv

Võrgu läbilaskevõime piiramisel pakettide arv väheneb (Joonis 52). Üleslüli ja allalüli pakettide arvud ei ole kahjuks üles märgitud. Oletada võib, et võrgu läbilaskevõime vähendamisel üleslüli pakettide arv väheneb kõikide protokollide korral. Ning allalüli pakettide arv väheneb kõikide transpordiks kasutatavate protokollide korral. TCP protokoll saadab välja vähem pakette, kui saabub vähem kinnitusi, UDP kasutamisel RealPlayer küsib saabumata jäänud pakette uuesti, millest tekib üleslüli pakettide arvu suurenemine. Testitud läbilaskevõimete juures RealPlayer pakette uuesti ei küsinud.



Joonis 53. Üleslüli läbilaskevõime vs pakettide arv.

Pakettide arv kokku väheneb iga protokollide korral üleslüli läbilaskevõime vähendamisel. Üleslüli ja allalüli pakettide arvud ei ole kahjuks üles märgitud. Oletada võib, et üleslüli läbilaskevõime vähendamisel kõikide protokollide korral üleslüli ja allalüli pakettide arvud vähenevad, aga natuke vähem kui eelmisel juhul. UDP korral võib aga allalüli pakettide arv samaks jääda.



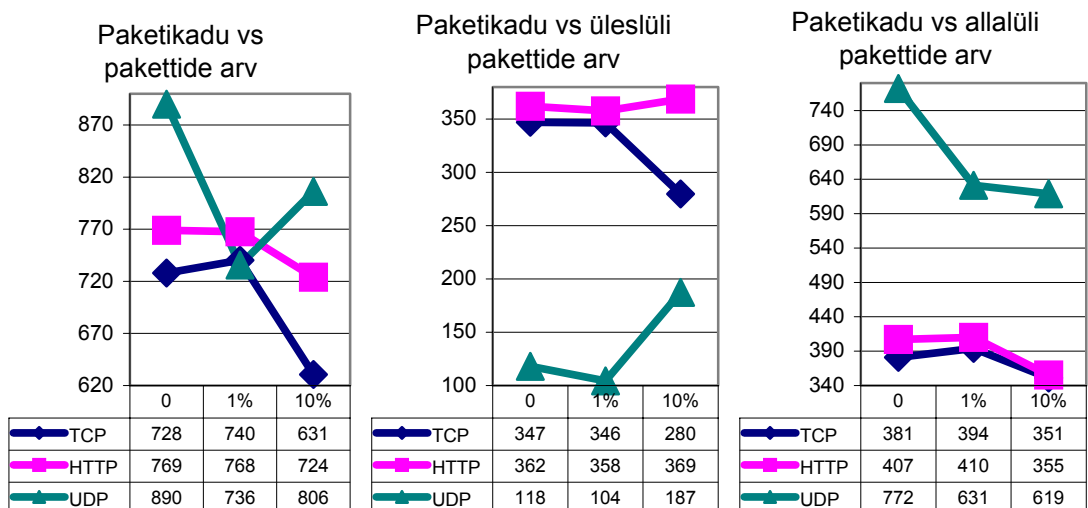
Joonis 54. Allalüli läbilaskevõime vs pakettide arv. Allalüli läbilaskevõime vs üleslüli pakettide arv. Allalüli läbilaskevõime vs allalüli pakettide arv.

Allalüli läbilaskevõime vähenemisel UDP korral üleslüli pakettide arv suureneb ja teiste protokollide korral väheneb. Allalüli läbilaskevõime väheneb kõikide transpordiprotokollide korral.

Läbilaskevõime – kokkuvõte

Läbilaskevõime ja üleslüli läbilaskevõime korral on mõlemate eraldi suundade paketid kokku lugemata. On teada mõlema suuna pakettide koguarvu muutus ja see liigub vähenemise suunas. Allalüli läbilaskevõime vaatlemisel ja RealPlayeri statistika põhjal võib oletada TCP ja HTTP korral üleslüli ja allalüli pakettide arvu vähenemist, UDP korral allalüli pakettide arvu väiksemat vähenemist ja üleslüli pakettide arvu samaks jäämist või väikest suurenemist.

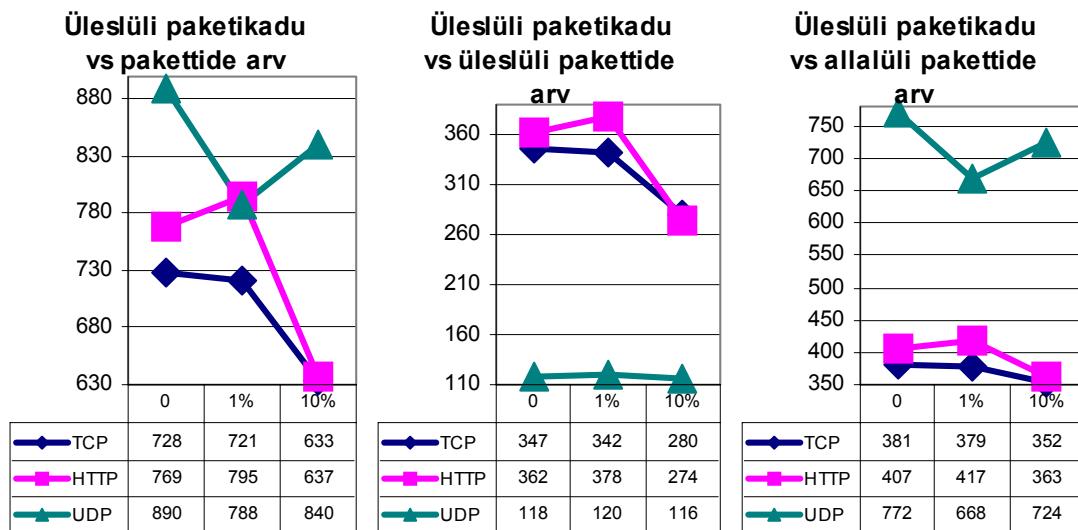
2.3.5.3 Paketikadu



Joonis 55. Paketikadu vs pakettide arv. Paketikadu vs üleslüli pakettide arv. Paketikadu vs allalüli pakettide arv.

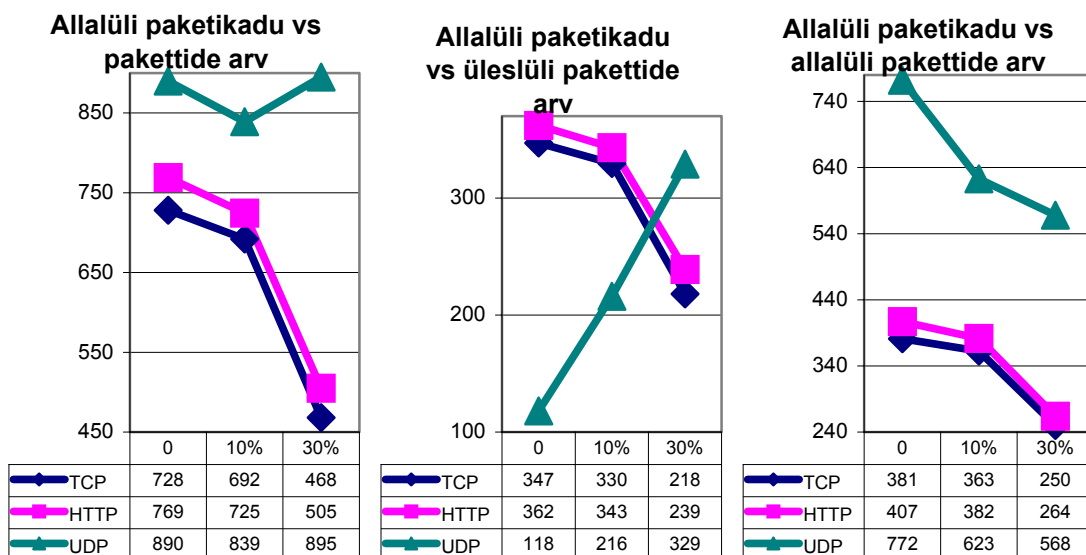
Paketikao suurenemisel TCP ja HTTP korral pakettide arv väheneb. HTTP korral jäi üleslüli pakettide arv samaks. UDP korral üleslüli pakettide arv suureneb ja allalüli

pakettide arv väheneb. Üleslüli pakettide arv on UDP korral väga palju suurenenud, sest RealPlayer on puudujäänud pakette uuesti küsinud paketikao 10% korral 10 korda rohkem kui paketikao 1% korral.



Joonis 56. Üleslüli paketikadu vs pakettide arv. Üleslüli paketikadu vs üleslüli pakettide arv. Üleslüli paketikadu vs allalüli pakettide arv.

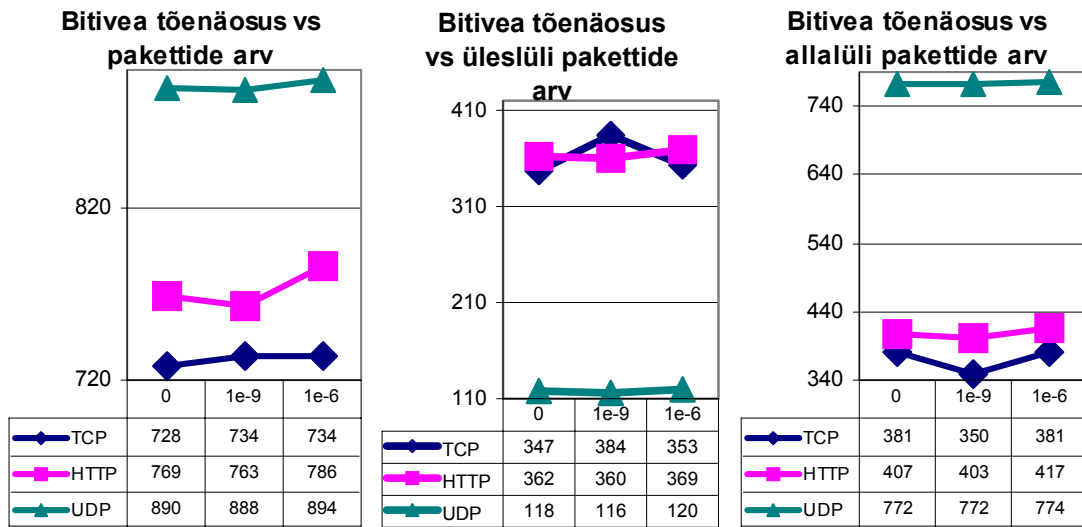
TCP ja HTTP korral pakettide arv väheneb mõlemal suunal, HTTP korral üleslüli pakettide arv jääb samaks. Paketikao 1% korral UDP allalüli andmehulk on väiksem kui paketikao 10% korral sellest, et paketikao 1% korral RealPlayer ei lülitanud kodeeritud andmevoo kiiruselt 292,4 ümber väiksemale kiirusele.



Joonis 57. Allalüli paketikadu vs pakettide arv. Allalüli paketikadu vs üleslüli pakettide arv. Allalüli paketikadu vs allalüli pakettide arv.

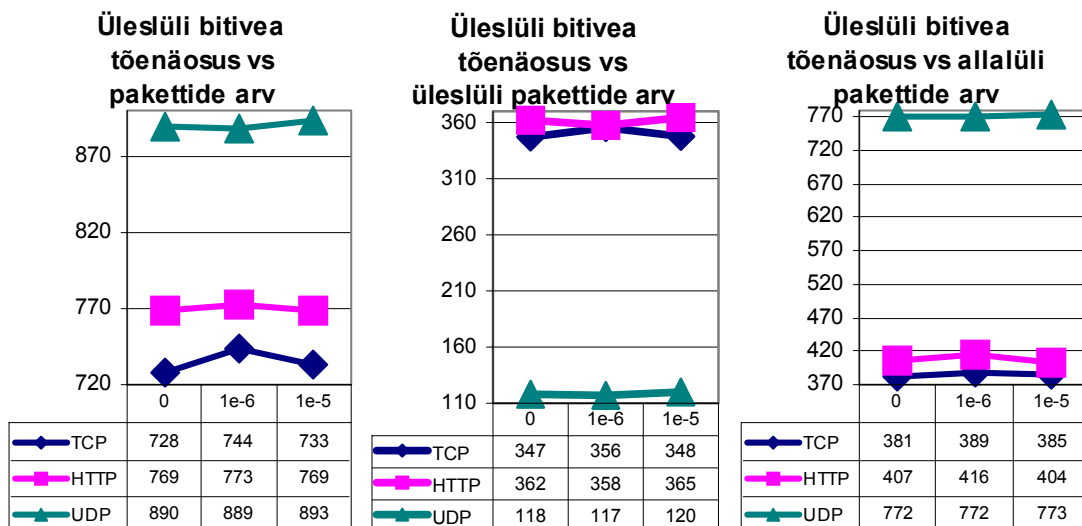
Allalüli paketikao suurenemisel kogu pakettide arv väheneb TCP ja HTTP korral, UDP korral aga suureneb. See suurenemine tekib üleslüli pakettide arvust. UDP korral, kui võrgu pakettidest kolmandik kadus, siis RealPlayer küsis teist korda ligi 200 paketti ja võttis neist vastu natuke üle 100.

2.3.5.4 Bitivea tõenäosus



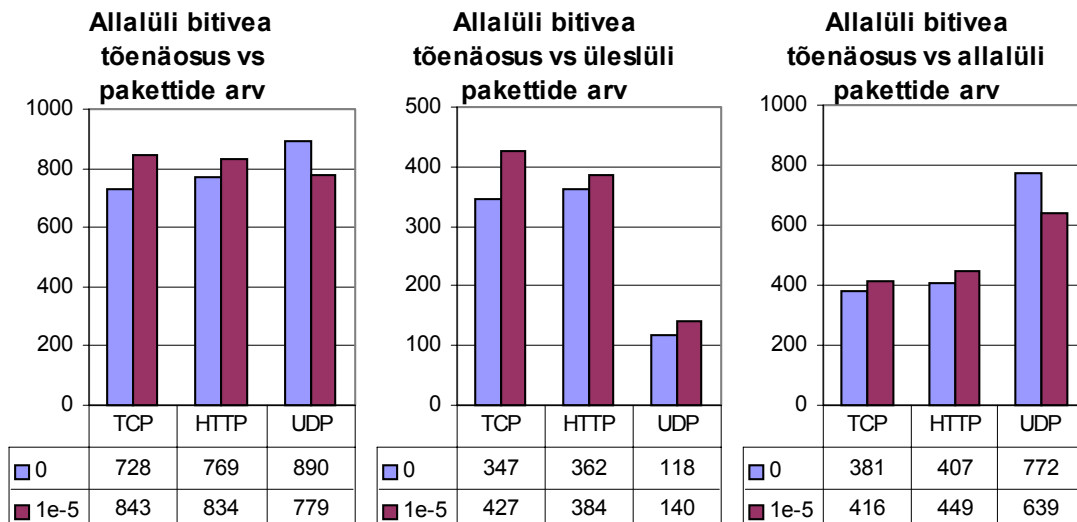
Joonis 58. Bitivea tõenäosus vs pakettide arv. Bitivea tõenäosus vs üleslüli pakettide arv. Bitivea tõenäosus vs allalüli pakettide arv.

Mõju ei avalda.



Joonis 59. Üleslüli bitivea tõenäosus vs pakettide arv. Üleslüli bitivea tõenäosus vs üleslüli pakettide arv. Üleslüli bitivea tõenäosus vs allalüli pakettide arv.

Mõju ei avalda.



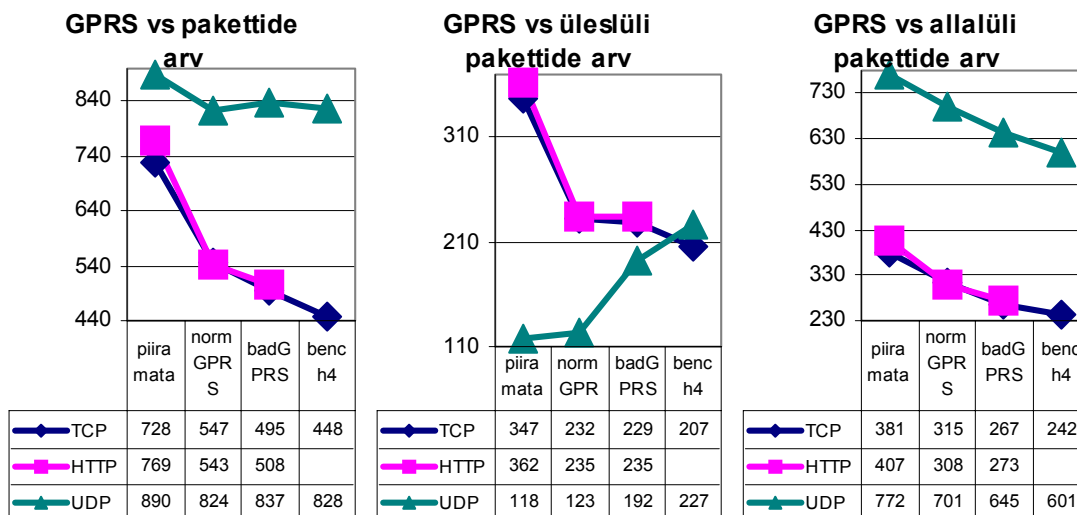
Joonis 60. Allalüli bitivea tõenäosus vs pakettide arv. Allalüli bitivea tõenäosus vs üleslüli pakettide arv. Allalüli bitivea tõenäosus vs allalüli pakettide arv.

Mõju ei avalda.

Bitivea tõenäosus – kokkuvõte

Bitivea tõenäosuse muutmine ei ole väga suurt mõju pakettide arvule avaldanud. On märgata väga vähest pakettide arvu suurenemist, sest bitivea tõenäosus rikub ära vähe pakette.

2.3.5.5 GPRS



Joonis 61. GPRS vs pakettide arv. GPRS vs üleslüli pakettide arv. GPRS vs allalüli pakettide arv.

Üldiselt pakettide arv väheneb. Üleslülis UDP korral pakettide arv suureneb.

2.3.5.6 Pakettide arv – kokkuvõte

Allalüli pakettide arv ja pakettide arv kokku on alati UDP korral suurem kui TCP ja HTTP korral ning enamasti see ka jääb nii erinevate võrgutingimuste kitsendamise korral. Üleslüli pakettide arv on UDP korral jälle TCP ja HTTP omast suurem ja

samuti jääb enamasti suuremaks võrgutingimuste kitsendamise korral. TCP ja HTTP pakettide arvud muutuvad üksteisele sarnaselt.

Võrgutingimuste kitsendamine vähendab või jätab samaks pakettide arvu allalülis ja kokku, üleslüli korral UDP korral enamasti pakettide arv suureneb.

Kõige väiksemat mõju avaldab bitivea tõenäosus. Suuremat mõju avaldavad TCP ja HTTP korral latentsus ja läbilaskevõime.

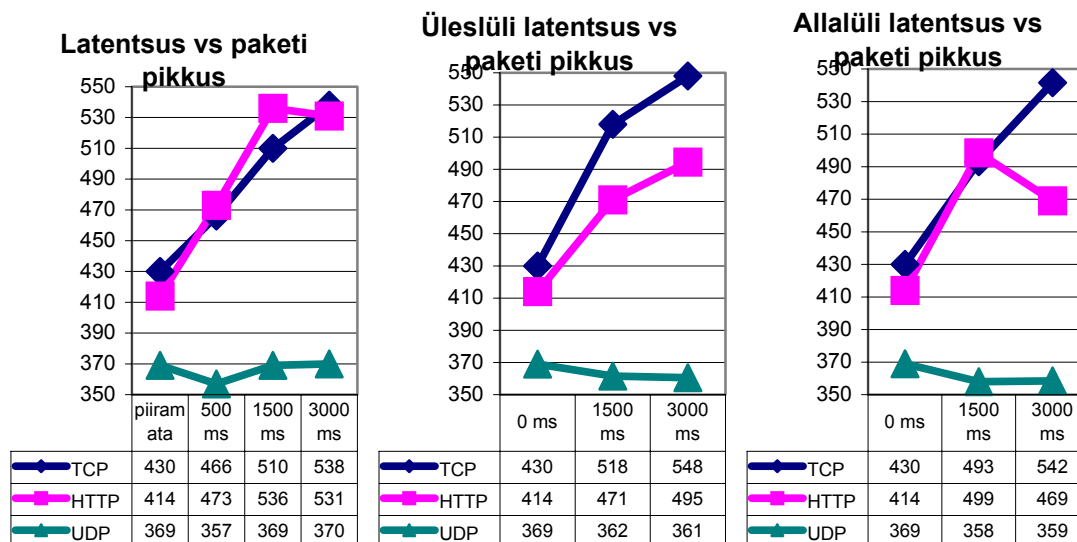
2.3.6 Paketi pikkus

Pikemad paketid kindlustavad korraga suurema andmehulga kättesaamise ja võrguressursside parema ärakasutamise. Kuid reaalaja rakendusele sobivad rohkem lühikesed paketid, mis saabuvad kohale kiiremini. Pika paketi saabumine võib nõuda seadmelt liiga suurt mälu mahtu. Väikeste pakettide saatmine võib muuta rakenduse pikatoimeliseks. Lühemate pakettide kasutamise korral on suurem osakaal ballastinfo nagu paketipäised, mille eest kasutaja maksab samuti.

Paketi keskmise pikkuse arvutas Epan. See sisaldab kõikide võrgukaablis mõlemas suunas liikunud pakettide keskmist pikkust. Edaspidi nimetame seda paketi pikkuseks ja mõõdame kilobaitides (kB).

Graafikutel on horisontaalteljel võrguparameetri väärtus võrgutingimuste halvenemise suunas ja vertikaalteljel on paketi pikkus baitides (B). Vasakpoolisel graafikul on võrguparameetri väärtus, mis piirab üleslüli ja allalüli, keskmisel graafikul võrguparameeter piirab ainult üleslüli ja allalüli on piiramata ning parempoolisel graafikul võrguparameeter piirab ainult allalüli ja üleslüli on piiramata.

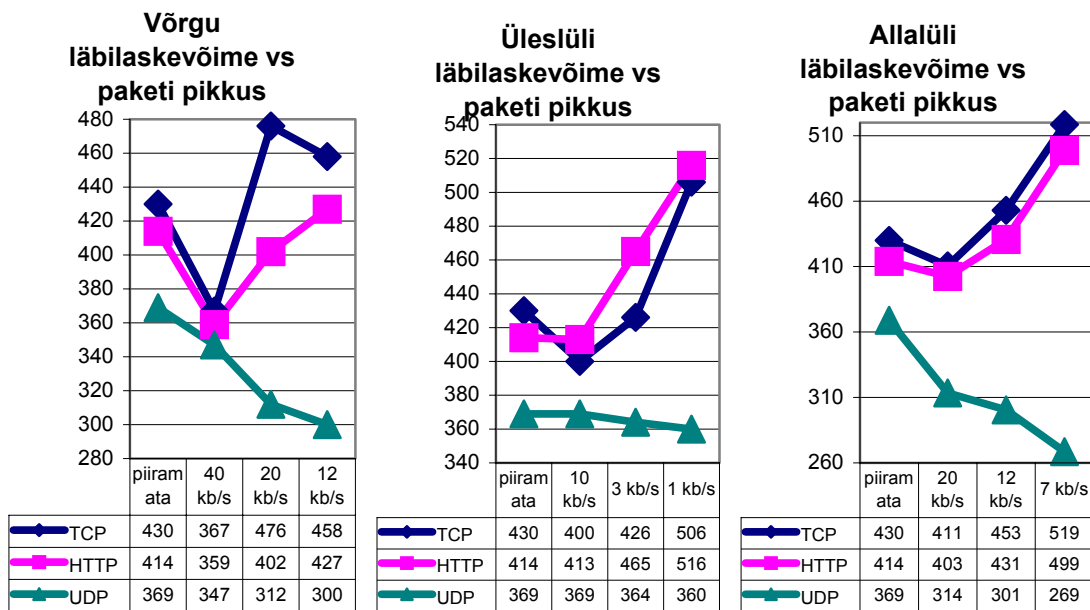
2.3.6.1 Latentsus



Joonis 62. Latentsus vs paketi pikkus. Üleslüli latentsus vs paketi pikkus. Allalüli latentsus vs paketi pikkus.

Latentsus on UDP korral paketi pikkusele mõju avaldamata jätnud, aga TCP ja HTTP korral paketi pikkus suureneb latentsuse suurenemise korral (Joonis 62). Üleslüli latentsus ja allalüli latentsus mõjutavad TCP ja UDP paketi pikkust ühtviisi, HTTP korral allalülis esialgu paketi pikkus suureneb, siis väheneb.

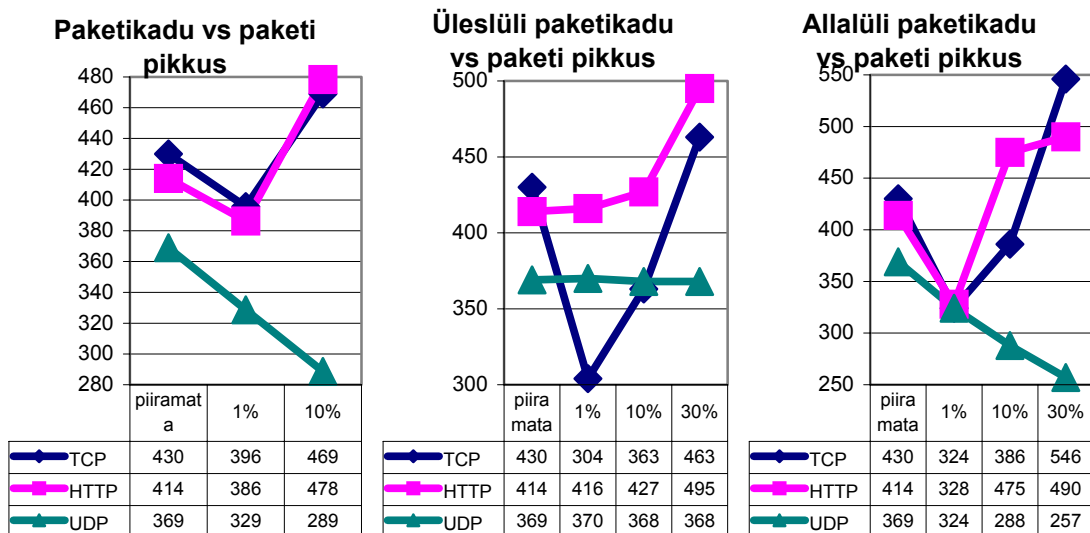
2.3.6.2 Läbilaskevõime



Joonis 63. Läbilaskevõime vs paketi pikkus. Üleslüli läbilaskevõime vs paketi pikkus. Allalüli läbilaskevõime vs paketi pikkus.

Läbilaskevõime vähenemine mõjutab paketi keskmist pikkust TCP ja HTTP korral piknemise suunas ja UDP korral lühenemise suunas. (Joonis 63)

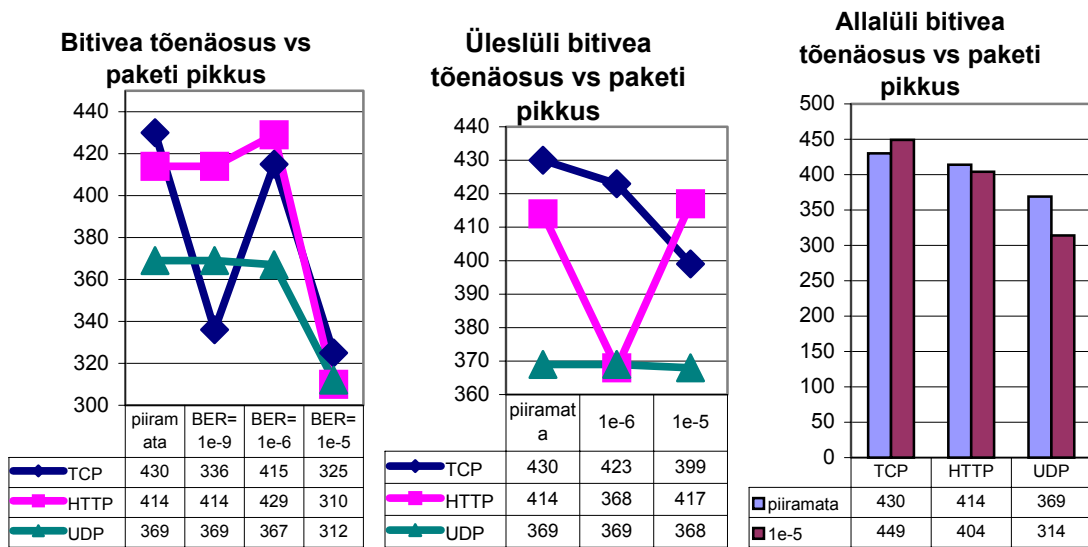
2.3.6.3 Paketikadu



Joonis 64. Paketikadu vs paketi pikkus. Üleslüli paketikadu vs paketi pikkus. Allalüli paketikadu vs paketi pikkus.

Üleslüli paketikadu ei mõjuta UDP korral paketi keskmist pikkust (Joonis 64 keskmine graafik), allalüli paketikadu mõjutab UDP paketi pikkust paketi lühenemise suunas (Joonis 64 parempoolne graafik). HTTP korral üleslüli paketikao korral pakett pikeneb monotoonselt, allalüli paketikao suurenemise korral paketi pikkus väheneb 1% juures, et siis suurenda edasi paketikao suurenedes. TCP korral käitub paketi pikkus sarnaselt nii üleslüli kui allalüli paketikao korral.

2.3.6.4 Bitivea tõenäosus

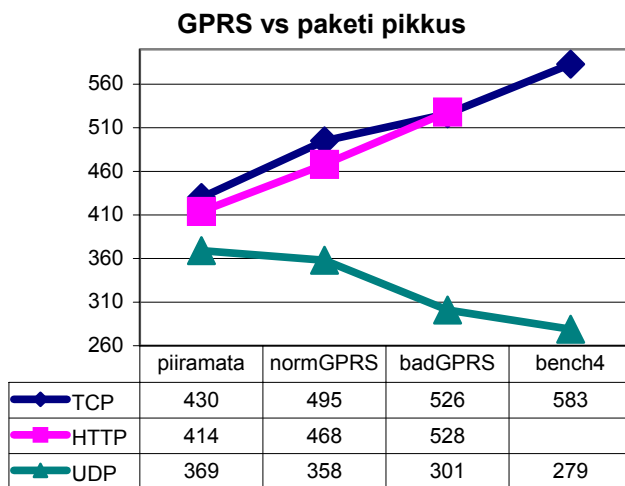


Joonis 65. Bitivea tõenäosus vs paketi pikkus. Üleslüli bitivea tõenäosus vs paketi pikkus. Allalüli bitivea tõenäosus vs paketi pikkus.

Bitivea tõenäosus hakkab paketi pikkust vähendama alles kriitilise piiri 10^{-4} (tähistatud $1e-4$) lähenemisel, selle piiri ületamisel ühenduse loomine juba ei õnnestu. Ühendus luuakse kasutades TCP protokollit.

Paketi pikkuse suur muutus (u 100 B) erinevatel bitivea tõenäosustel (Joonis 65 vasakpoolne graafik, TCP) on sellest, mitu paketti RealPlayer kätte saada soovib. TCP ja HTTP korral kui paketi pikkus oli u 400 B, siis RealPlayer pakettide arv oli u 700, ja kui paketi pikkus oli u 300 B, siis RealPlayer pakettide arv oli u 600.

2.3.6.5 GPRS



Joonis 66. GPRS vs paketi pikkus

Erinevad GPRS tingimused suudavad TCP ja HTTP korral paketi pikkust suurendada, UDP korral aga vähendada.

2.3.6.6 Paketi pikkus – kokkuvõte

TCP ja HTTP korral alati keskmine paketi pikkus suureneb, kuigi mõne võrguparameetri vähesel muutmisel paketi pikkus hoopiski väheneb. Üleslülü paketikadu pikendab HTTP korral paketti rohkem kui TCP korral. UDP korral suudab paketi pikkust vähenemise suunas mõjutada allalülü paketikadu ja läbilaskevõime.

Kõige rohkem mõjutab TCP ja HTTP korral paketi pikkust suurenemise suunas latentsus. Bitivea tõenäosuse suurenemisel paketi pikkus vähenes hoopis.

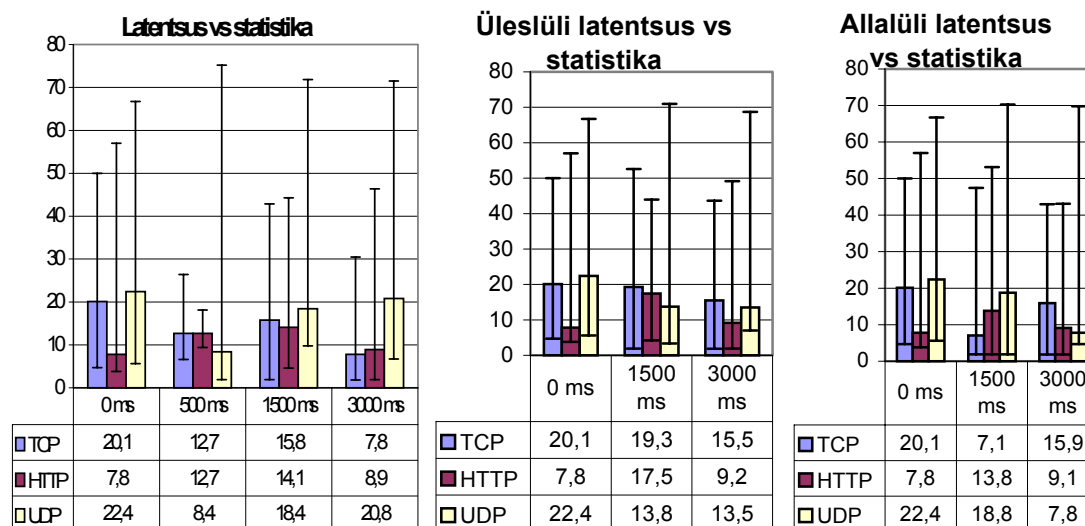
2.3.7 RealPlayeri edastuskiiruse statistika

Edastuskiiruse statistika annab kasutajale teada võrgu olukorrast.

RealPlayer annab soovitud pakettide arvu, klipi jooksul arvatatud min, max ja keskmise edastuskiiruse. Kui kasutame UDP protokollit, siis saame ka kadunud pakettide arvu teada. TCP ja HTTP korral RealPlayer paketikaost midagi teada ei saa, sest transpordiprotokoll ise küsib saabumata paketeid uuesti ja annab need edasi RealPlayerile.

Graafikutel on horisontaalteljel võrguparameetri väärtus võrgutingimuste halvenemise suunas ja vertikaalteljel on edastuskiirused (väikseim, suurim ja keskmine) kilobitti sekundis (kb/s). Vasakpoolisel graafikul on võrguparameetri väärtus, mis piirab üleslülü ja allalülü, keskmisel graafikul võrguparameeter piirab ainult üleslülü ja allalülü on piiramata, ning parempoolisel graafikul võrguparameeter piirab ainult allalülü ja üleslülü on piiramata.

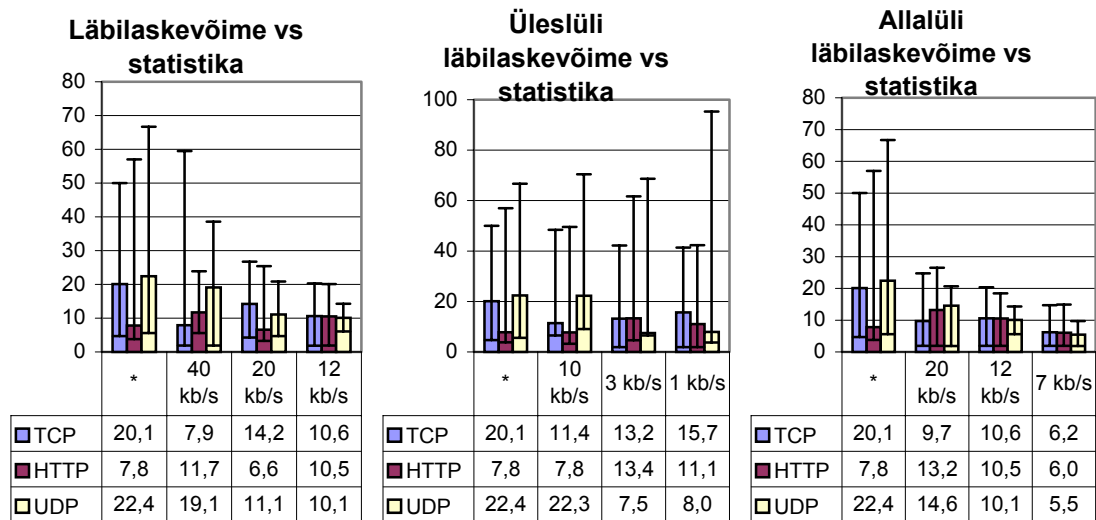
2.3.7.1 Latentsus



Joonis 67. Latentsus vs statistika. Üleslülü latentsus vs statistika. Allalülü latentsus vs statistika.

Latentsuse mõju on väga erinev. Latentsus on mõjutanud TCP ja HTTP suurimat edastuskiiruse väärtust. UDP suurim väärtus on teistest suuremaks jäänud ka latentsuse suurenedes. UDP korral keskmine väärtus üleslülü latentsuse ja allalülü latentsuse korral väheneb. HTTP korral saavutatakse kõige suurem keskmine väärtus latentsuse 1500 ms korral kõigil juhtudel. TCP suurim väärtus on enamasti kõige väiksem teiste protokollide suurimatest väärtustest.

2.3.7.2 Läbilaskevõime



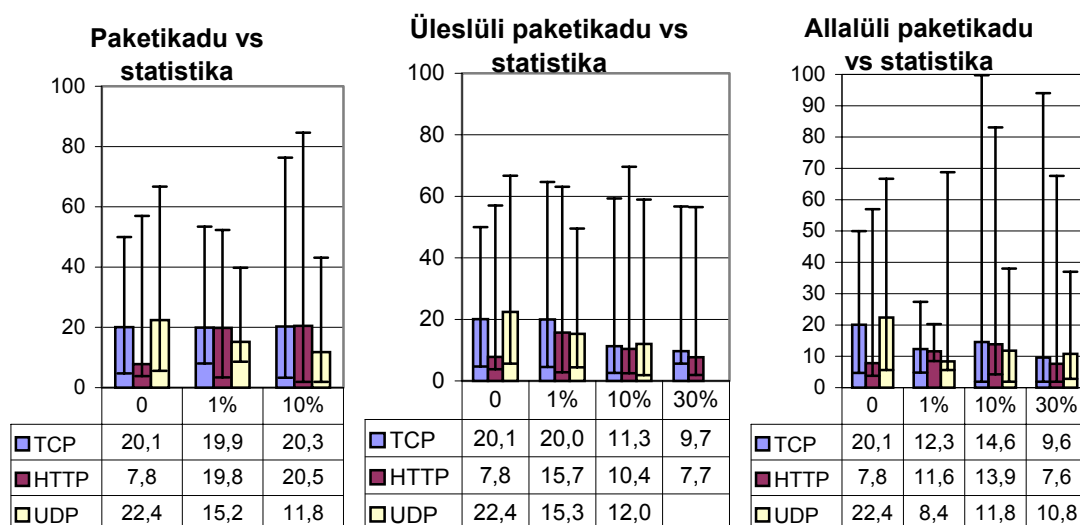
Joonis 68. Läbilaskevõime vs statistika. Üleslüli läbilaskevõime vs statistika. Allalüli läbilaskevõime vs statistika.

Läbilaskevõime vähendamine on vähendanud kõiki edastuskiiruse väärtusi. Läbilaskevõime 12 kb/s korral on kõikide transpordiprotokollide edastuskiiruse statistika keskmised väärtused võrdsed.

Üleslüli läbilaskevõime piiramine palju mõju statistikale ei avalda. UDP korral on suurim väärtus isegi kõige suurem teiste transpordiprotokollide omadest. Keskmise väärtus väheneb TCP ja UDP korral, HTTP korral suureneb natuke läbilaskevõime vähendamisel.

Allalüli läbilaskevõime piiramine on väga suurt mõju avaldanud suurimatele väärtustele ja suurt mõju avaldanud keskmisele väärtusele. Rohkem mõjutab UDP korral läbilaskevõimet kui teiste transpordiprotokollide korral.

2.3.7.3 Paketikadu

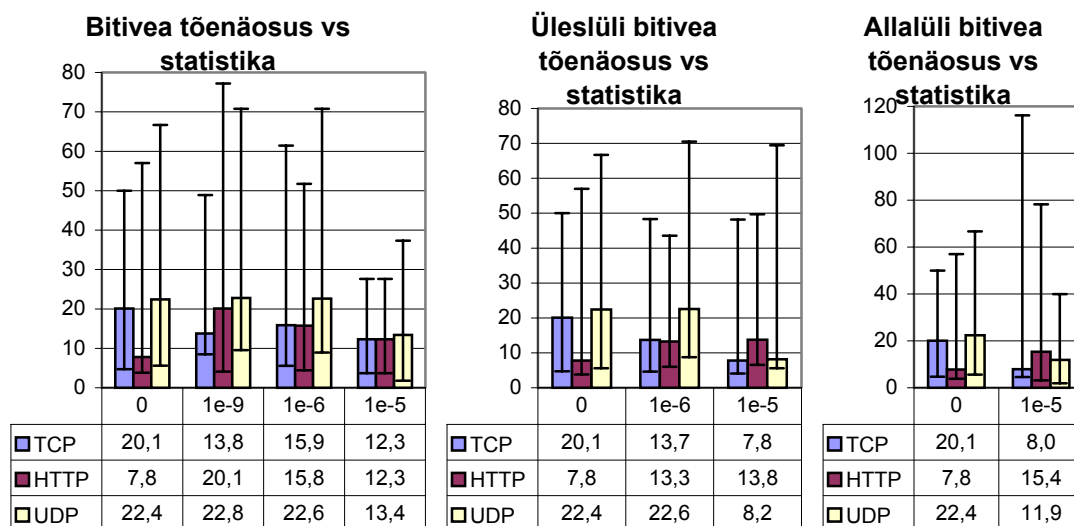


Joonis 69. Paketikadu vs statistika. Üleslüli paketikadu vs statistika. Allalüli paketikadu vs statistika.

Keskmesid väärtused jäävad üsna madalale võrreldes suurimate väärtustega.

Paketikao korral TCP statistika keskmine jääb samaks, HTTP korral suureneb, UDP korral väheneb (Joonis 69 vasakpoolne graafik). Üleslüli paketikao korral TCP korral esialgu jääb samaks, suure paketikao korral keskmine väheneb (Joonis 69 keskmine graafik). UDP keskmine väheneb üleslüli paketikao suurenemise korral. Allalüli paketikao korral on suurimad väärtused väiksemad kui teistel paketikao juhtudel (Joonis 69 parempoolne graafik).

2.3.7.4 Bitivea tõenäosus



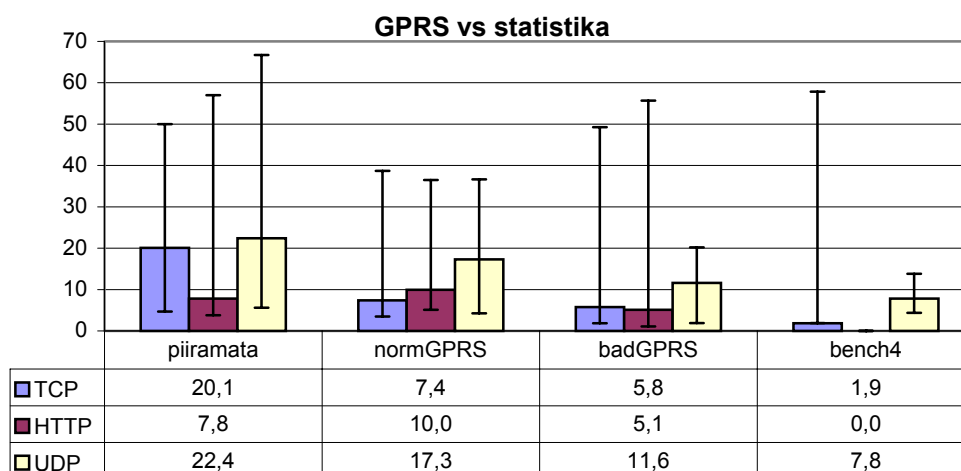
Joonis 70. Bitivea tõenäosus vs statistika. Üleslüli bitivea tõenäosus vs statistika. Allalüli bitivea tõenäosus vs statistika.

Bitivea tõenäosuse suurenedes TCP korral esialgu edastuskiiruse keskmine väheneb, siis natuke suureneb; HTTP korral keskmine esialgu suureneb, siis väheneb ja UDP korral esialgu jääb samaks, siis väheneb (Joonis 70 vasakpoolne graafik). Bitivea tõenäosuse 10^{-5} (tähistatud 1e-5) korral on edastuskiiruse suurimad väärtused palju väiksemad kui väiksema bitivea tõenäosuse korral.

Üleslüli bitivea tõenäosus pole mõjutanud suurimat väärtust. Keskmine väärtus TCP korral väheneb monotoonselt, HTTP korral natuke suureneb, UDP korral väheneb alles suure bitivea tõenäosuse korral (Joonis 70 keskmine graafik).

Allalüli bitivea tõenäosus on TCP korral suurendanud suurimat väärtust ja natuke vähem suurendanud suurimat väärtust HTTP korral. UDP korral on suurim väärtus vähenenud. Keskmine väärtus on TCP ja UDP korral vähenenud, HTTP korral suurenenud (Joonis 70 parempoolne graafik).

2.3.7.5 GPRS



Joonis 71. GPRS vs statistika.

Testid näitavad, et HTTP korral edastuskiiruse keskmine suureneb võrreldes piiranguteta võrguga, kuid kõige suurem keskmine saavutatakse iga võrgutingimuste juures UDP korral (Joonis 71). Võrgutingimuste bench4 korral HTTP katse ei õnnestunud 5 korral, nii siin tulemust kirja ei saanudki.

2.3.7.6 RealPlayeri edastuskiiruse statistika – kokkuvõte

Läbilaskevõime ja allalüli läbilaskevõime on kõige rohkem statistikat mõjutanud. Teiste parameetrite korral on keskmine vähenenud kuni 10 kb/s võrra. Teiste parameetrite ja läbilaskevõime korral on max väärtused samaks jäänud või ka suurenenud.

Suurimate väärtuste kohta ei saa kindlat muutuse tendentsi näidata, välja arvatud nende vähenemine läbilaskevõime ja allalüli läbilaskevõime vähendamisel. Vähima väärtuse kõige väiksem väärtus on 1,9 kb/s ja seda oli võimalik saavutada igasugustel võrgutingimustel. Vähima väärtuse hulka ei loetud ajahetked, kui saabus andmeid 0 kb/s, aga keskmise väärtuse arvutamisel läksid need ajahetked arvesse. Keskmise väärtuse vähenemist võrgutingimuste halvenemise korral on näha üleslüli latentsuse suurendamisel TCP ja UDP korral, allalüli latentsuse suurendamisel UDP korral, läbilaskevõime, üleslüli läbilaskevõime ja allalüli läbilaskevõime vähendamisel UDP korral, paketikao suurendamisel UDP korral, üleslüli paketikao suurendamisel ja GPRS tingimuste korral. TCP ja HTTP korral transpordiprotokoll reguleerib pakettide saabumist rohkem kui UDP korral

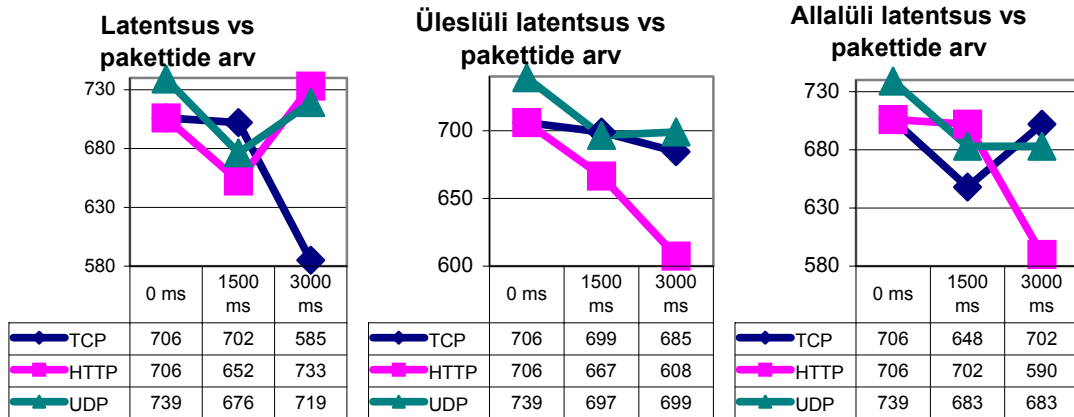
2.3.8 RealPlayer pakettide arv

Olenevalt kodeeritud andmevoo kiirusest (*encode rate*) vajab RealPlayer erineva arvu pakette. Neid ta loeb ka ja kasutaja saab neid näha klipi vaatamise ajal menüüst *View->Statistics*.

Graafikute horisontaalteljel on võrguparameetri väärtus võrgutingimuste halvenemise suunas ja vertikaalteljel on pakettide arv (ühikuta). Vasakpoolsel graafikul on võrguparameetri väärtus, mis piirab üleslüli ja allalüli, keskmisel graafikul

võrguparameeter piirab ainult üleslüli ja allalüli on piiramata, ning parempoolsele graafikule võrguparameeter piirab ainult allalüli ja üleslüli on piiramata.

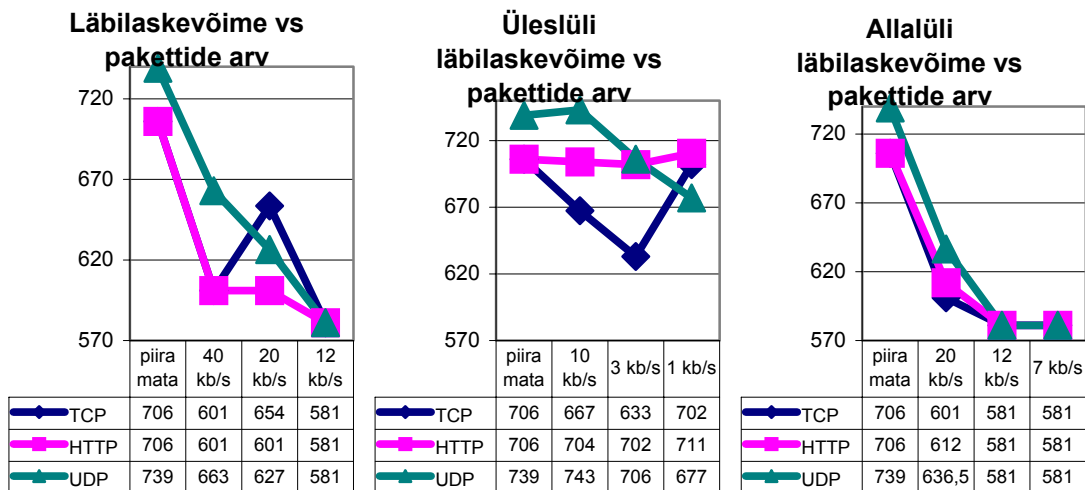
2.3.8.1 Latentsus



Joonis 72. Latentsus vs pakettide arv. Üleslüli latentsus vs pakettide arv. Allalüli latentsus vs pakettide arv.

Latentsuse suurenedes TCP korral RealPlayer on soovinud saada vähem pakette (Joonis 72). Siis oli klipi vaatamise lõpuks kodeeritud kiirus 12 kb/s, teiste korral oli 20 kb/s. HTTP korral tuleb väiksem pakettide arv latentsuse 1500 ms juures (Joonis 72 vasakpoolne graafik) sellest, et siis valiti andmevoo edastuskiiruseks 15 kb/s, aga üleslüli latentsuse 1500 ms ja allalüli latentsuse 1500 ms juures valiti andmevoo edastuskiiruseks 20 kb/s (Joonis 72 keskmine ja parempoolne graafik).

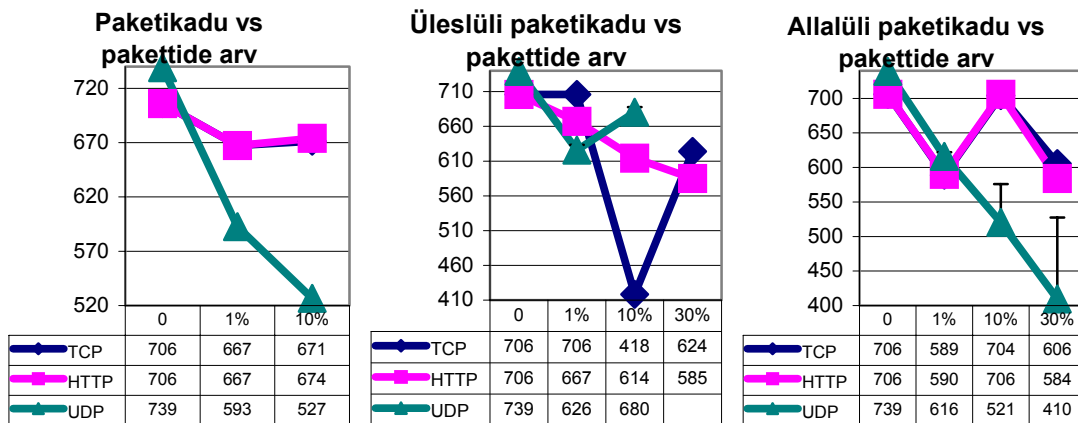
2.3.8.2 Läbilaskevõime



Joonis 73. Läbilaskevõime vs pakettide arv. Üleslüli läbilaskevõime vs pakettide arv. Allalüli läbilaskevõime vs pakettide arv.

Läbilaskevõime piiramisel RealPlayer soovib järjest vähem pakette kätte saada. UDP korral ühelgi juhul RealPlayer ei tuvastanud puudujäävaid pakette. Üleslüli läbilaskevõime piiramisel HTTP korral on pakettide arv samaks jäänud.

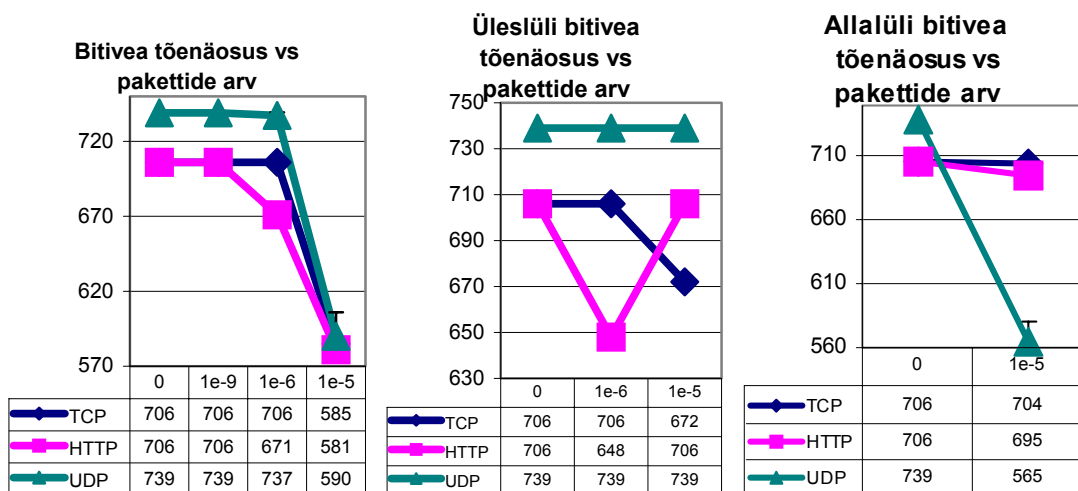
2.3.8.3 Paketikadu



Joonis 74. Paketikadu vs pakettide arv. Üleslüli paketikadu vs pakettide arv. Allalüli paketikadu vs pakettide arv.

Paketikao suurenemine on RealPlayerit sundinud vajama vähem pakette (Joonis 74 vasakpoolne graafik). Üleslüli paketikadu on UDP korral erinevalt mõjunud ja allalüli paketikadu on TCP ja HTTP korral mõjunud erinevalt. Allalüli paketikao suurenemisel UDP korral RealPlayeri soovitud pakettide arv väheneb ja on näidatud tegelikult kätte saadud pakettide arvu juures veana (Joonis 74 parempoolne graafik, UDP).

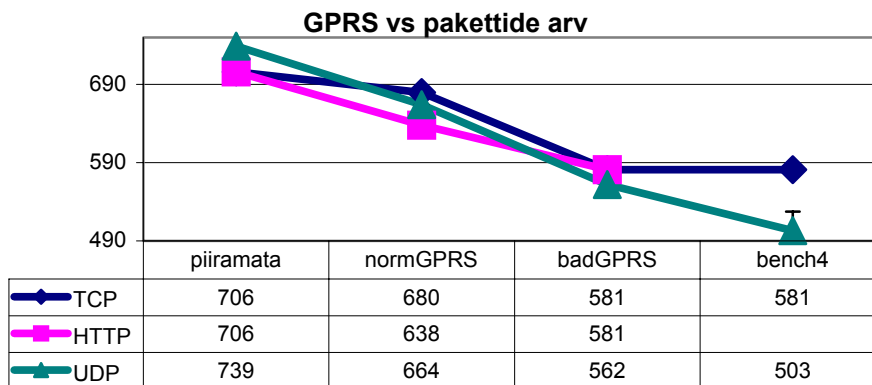
2.3.8.4 Bitivea tõenäosus



Joonis 75. Bitivea tõenäosus vs pakettide arv. Üleslüli bitivea tõenäosus vs pakettide arv. Allalüli bitivea tõenäosus vs pakettide arv.

Väga suur bitivea tõenäosus vähendab RealPlayeri poolt vajatavate pakettide arvu järsult (Joonis 75 vasakpoolne graafik). Üleslüli bitivea tõenäosus ei mõjuta UDP korral ja allalüli bitivea tõenäosus ei mõjuta TCP ja HTTP korral RealPlayeri poolt vajatavate pakettide arvu.

2.3.8.5 GPRS



Joonis 76. GPRS vs pakettide arv

UDP korral väheneb RealPlayeri soovitud pakettide arv ühtlasema kiirusega kui teiste protokollide korral.

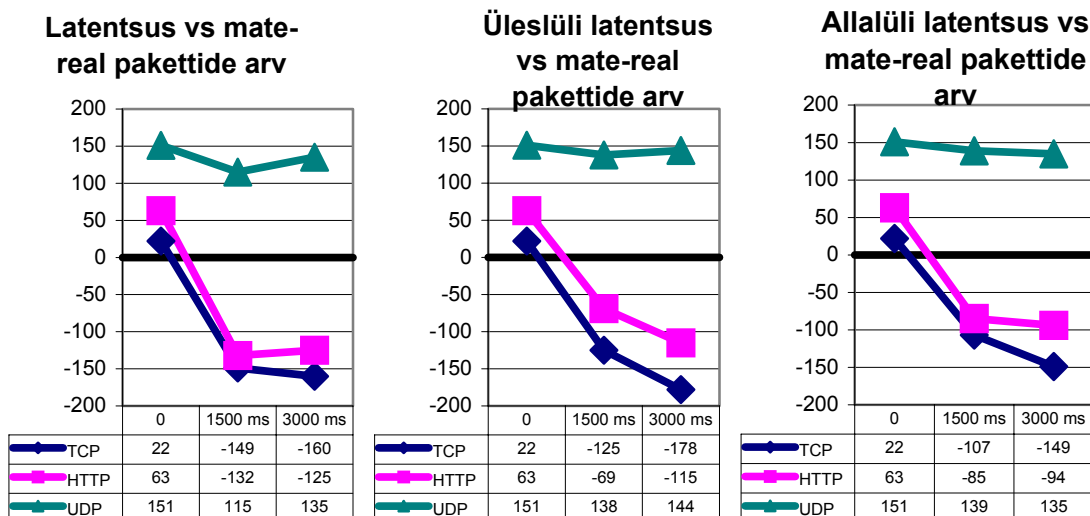
2.3.9 Pakettide arvu võrdlus

Vaatame, kuidas erinevad RealPlayeri statistika ja võrgu statistika (MATE) pakettide arvu muutumise poolest. Siin selgub ballastinfo osakaalu muutus vastavalt võrgutingimuste muutusele. Samuti selgub, kas kasutaja võib saada õiget ülevaadet võrgu olukorrast.

MATE loendatud pakettide arvust on lahutatud RealPlayeri pakettide arv (tähistatud mate-real).

Graafikutel on horisontaalteljel võrguparameetri väärtus võrgutingimuste halvenemise suunas ja vertikaalteljel on pakettide arv (ühikuta). Vasakpoolsel graafikul on võrguparameetri väärtus, mis piirab üleslüli ja allalüli, keskmisel graafikul võrguparameeter piirab ainult üleslüli ja allalüli on piiramata, ning parempoolsel graafikul võrguparameeter piirab ainult allalüli ja üleslüli on piiramata.

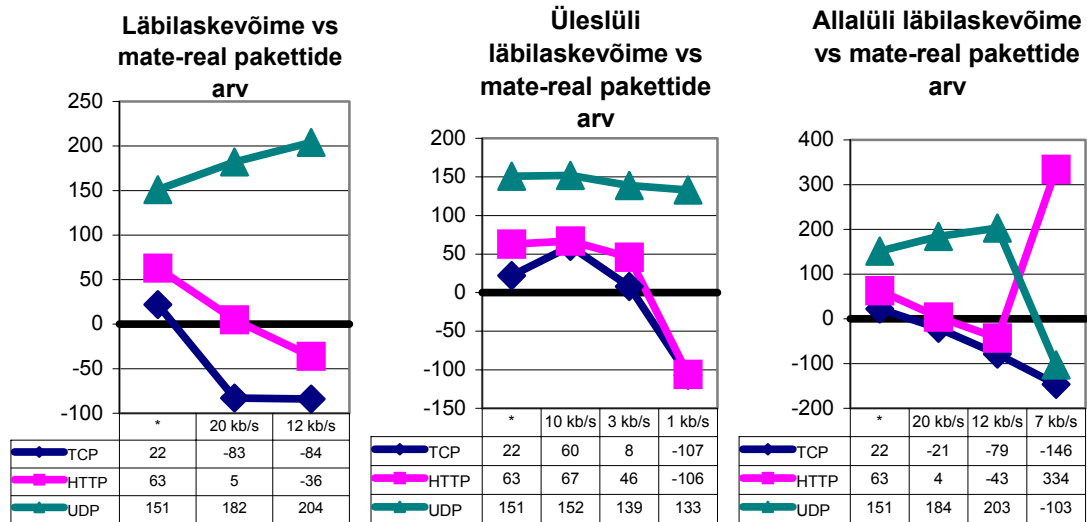
2.3.9.1 Latentsus



Joonis 77. Latentsus vs mate-real pakettide arv. Üleslüli latentsus vs mate-real pakettide arv. Allalüli latentsus vs mate-real pakettide arv.

Latentsuse suurenedes on pakettide arv vähenenud. TCP korral RealPlayeri statistikas on pakettide arv suurem kui võrgu statistikas, võidakse panna mitu RealPlayeri paketti ühe TCP paketi sisse. HTTP korral samuti latentsuse suurenedes jääb võrgu pakettide arv väiksemaks kui RealPlayeri pakettide arv. UDP korral alati on võrgu pakettide arv suurem kui RealPlayeri oma.

2.3.9.2 Läbilaskevõime

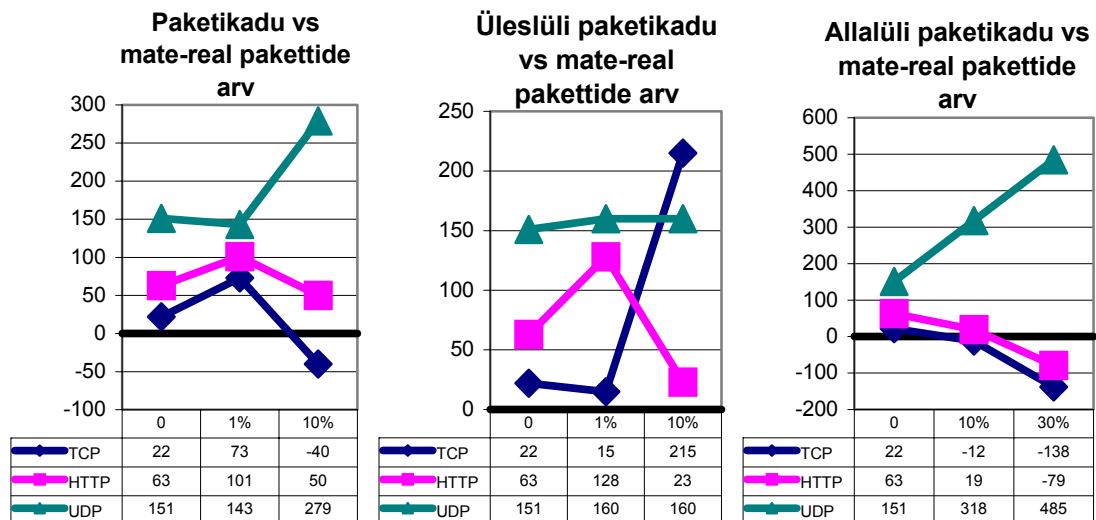


Joonis 78. Läbilaskevõime vs mate-real pakettide arv. Üleslüli läbilaskevõime vs mate-real pakettide arv. Allalüli läbilaskevõime vs mate-real pakettide arv.

Läbilaskevõime piiramisel TCP ja HTTP korral edastatakse võrgu kaudu vähemate pakettidega rohkem RealPlayeri pakette. UDP korral mitut RealPlayeri paketti ühe võrgupaketi sisse ei panda.

Üleslüli läbilaskevõime piiramisel pakitakse rohkem RealPlayeri pakette võrgu pakettide sisse.

2.3.9.3 Paketikadu



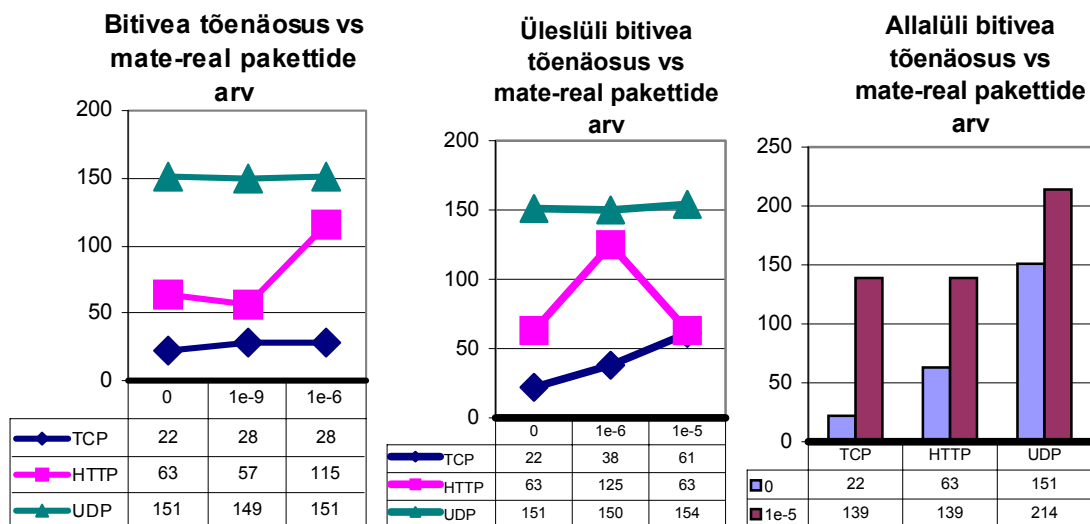
Joonis 79. Paketikadu vs mate-real pakettide arv. Üleslüli paketikadu vs mate-real pakettide arv. Allalüli paketikadu vs mate-real pakettide arv.

Paketikao suurenemise korral TCP ja HTTP korral esialgu võrgus liigub rohkem pakette, kui RealPlayer vajab, edasisel paketikao suurenemisel TCP korral pannakse jälle mitu RealPlayeri paketti ühe võrgupaketi sisse. UDP korral ei panda.

Üleslüli paketikao korral lihtsalt võrgus liigub rohkem pakette ja TCP korral see arv suureneb väga palju.

Allalüli paketikao suurenemisel nii TCP kui HTTP korral pannakse RealPlayeri paketid võrgupakettide sisse, UDP korral mitte (Joonis 79 parempoolne graafik).

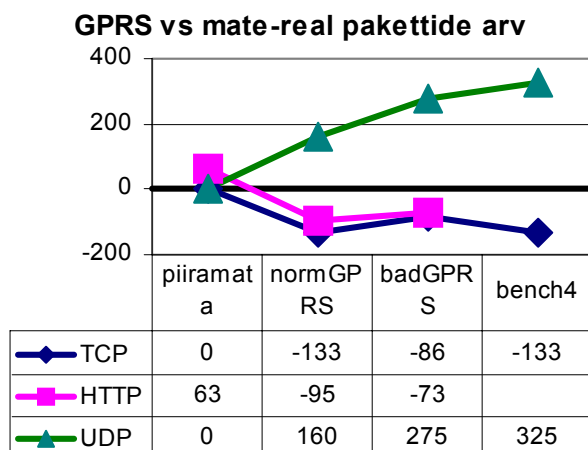
2.3.9.4 Bitivea tõenäosus



Joonis 80. Bitivea tõenäosus vs mate-real pakettide arv. Üleslüli bitivea tõenäosus vs mate-real pakettide arv. Allalüli bitivea tõenäosus vs mate-real pakettide arv.

Suurema bitivea tõenäosusega liigub võrgus natuke rohkem pakette kõikide protokollide korral.

2.3.9.5 GPRS



Joonis 81. GPRS vs mate-real pakettide arv

Erinevate GPRS tingimuste korral kasutades TCP ja HTTP protokolle pannakse mitu RealPlayeri paketti ühe võrgus liikuva paketi sisse, UDP kasutamisel ei panda.

2.3.9.6 Pakettide arvu võrdlus – kokkuvõte

Suurte üleslüli paketikadude ja bitivea tõenäosuste korral ning TCP ja HTTP transpordiprotokollide kasutamise korral ei panda mitut RealPlayer paketti ühe võrgus liikuva paketi sisse.

UDP korral ei panda kunagi mitut RealPlayer paketti ühe võrgus liikuva paketi sisse. UDP korral võrgus liikuvate pakettide arvu suurenemist võrreldes RealPlayeri pakettide arvuga on näha läbilaskevõime, paketikao, allalüli paketikao ja allalüli bitivea tõenäosuse korral.

Allalüli läbilaskevõime piiramine kuni alla poole vajamineva tingib TCP ja UDP korral mitme RealPlayeri paketi panekut võrgus liikuva paketi sisse. HTTP korral aga nii ei juhtunud, kuigi teistel võrgupiirangutel käitus sarnaselt TCP-ga.

2.3.10 Katsete kokkuvõte

Kuidas RealPlayeri statistika ühtib Epani ja MATE võrgu pealt kogutavate andmetega.

Latentsus

Latentsus mõjutab aegu, sest latentsuse põhjustab pakettide edastusel tekkiv viide. Pakettide arvu, andmehulga ja edastuskiiruse statistika juures TCP ja HTTP korral tulemused erinevad ja kindlat vähenemist määrata ei saa, sest see sõltub klipi andmevoo edastuskiiruse valikust ja selle muutmisest sessiooni ajal. UDP korral paketi pikkus ei muutu latentsuse suurenedes. See tähendab, et UDP on latentsuse suhtes tolerantsem, aga TCP ja HTTP arvestavad võimaliku võrgu ummistuse tekkega ja saadavad välja vähem pakette.

Läbilaskevõime

Läbilaskevõime ja allalüli läbilaskevõime piiramine pikendab aegu (klipi andmevoo edastuse lõpu ajad lähenevad ettemängimise lõpu ajale), vähendab andmehulki ja pakettide arvu (va UDP üleslüli andmehulk ja pakettide arv suurenevad) ja vähendab RealPlayeri statistika väärtusi. Paketi keskmine pikkus UDP korral kindlalt väheneb, teiste korral esialgu natuke väheneb ja siis suureneb isegi pikemaks kui oli piiramata läbilaskevõimega võrgus. Üleslüli läbilaskevõime piiramisel TCP ja HTTP korral ajad, pakettide arv ja üleslüli andmehulk muutuvad nii nagu allalüli läbilaskevõime piiramisel, UDP korral ajad pikenevad ühenduse loomise aja pikenedes võrra, andmehulk väheneb, üleslüli pakettide arv suureneb. Üleslüli läbilaskevõime piiramisel RealPlayeri pakettide arv kõikide transpordiprotokollide korral ning TCP ja HTTP andmehulgad sõltuvad valitud andmevoo edastuskiirusest kõikide protokollide korral. RealPlayeri pakettide arv muutub erinevalt (vastavalt valitud andmevoo edastuskiirusele) läbilaskevõime ja üleslüli läbilaskevõime piiramisel

See võib tulla TCP andmevoo juhtimise algoritmist nn “*slow start algorithm*”, kus edastamise akna suurus väheneb (võib saata mitu paketti ära ja siis oodata nende kinnitusi), kui kinnitused ei saabu teatud aja jooksul.

Paketikadu

Paketikadu pikendab pildi näitamise alguse ja koguaega kõige rohkem HTTP korral ja edastuse lõpu aega UDP korral. Üleslüli paketikadu (kuni 10%) ei pikendanud aegu üldse UDP korral, aga TCP ja HTTP korral pikendas. 30% üleslüli paketikao korral oli TCP korral aeg lähedane eelmistele, HTTP korral 2x pikem ja UDP korral klippi vaadata ei õnnestunud. Allalüli paketikao 30% korral oli kõige lühem aeg UDP korral,

pikim HTTP korral. TCP ja HTTP korral küsib transpordiprotokoll kadunud pakettid saatjalt uuesti ja need ei kajastu RealPlayeri saamata pakettide arvus, UDP korral RealPlayer küsib kadunud pakettid uuesti ja see kajastub ka RealPlayeri pakettide arvu statistikas. Üleslüli paketikadu 30% põhjustab UDP korral ühenduse katkemise või ühendust luua ei õnnestugi. Üleslülis liigub UDP korral umbes poole vähem pakette ja seetõttu võib mõne olulisema paketi mittesaabumine edastust mõjutada.

Bitivea tõenäosus

Bitivea tõenäosus on kõiki mõõdetavaid parameetreid mõjutanud vähe. See tuleb sellest, et bitiviga rikub ära vähe pakette.

Järeldused

Ajad sõltuvad eelkõige pakettide saabumisest. RealPlayeri andmevoo pakettide arv sõltub RealPlayeri andmevoo edastuskiiruse valikust. Andmevoo edastuskiiruse valik sõltub võrgust, aga tihti, kui üleslüli on piiratud ja allalüli on piiramata, valitakse väiksem andmevoo edastuskiirus kui võiks. RealPlayeri edastuskiiruse statistika arvud kogu klipi kohta ei anna alati infot võrgu olukorra kohta, seda saab paremini jooksvat statistikat jälgides. Ainult UDP korral keskmine edastuskiirus võrguparameetrite halvenemisel tavaliselt väheneb. TCP ja HTTP korral on keskmine paketi pikkus pikem kui UDP korral ning mitu RealPlayeri paketti pannakse ühe võrgus liikuva paketi sisse TCP ja HTTP korral, aga mitte UDP korral. Allalüli pakettide kadumine või rikkumine põhjustab üleslüli pakettide arvu suurenemise UDP korral. TCP ja HTTP korral andmevoo saatmine (allalüli) on rohkem mõjutatud ka võrgu üleslüli olukorrast kui UDP korral.

Nn asümmeetrilise sidekanali (üleslüli ja allalüli parameetrid erinevad) korral rohkem piiratud üleslüliga töötades transpordiprotokollidel jääb võrgu olukorrast vale mulje ja ei kasutata ära allalüli kõiki ressursse.

2.4 Majanduslik külg

Majandushinnangute andmisel lähtuti nõudest tagada katkestusteta ettemängimine.

Hinnakiri 21.02.2002:

EMT mobiilinterneti kõnehind 2,49 kr/min, kõne kestuse mõõtmise on 1 sekundi täpsusega (<http://www.emt.ee/est/html4/?smg=hinnad&schg=paketid.andmeside>)

EMT andmehulga järgi maksustamise hinnakirja järgi on andmehulga maksumus 40 kr/MB ja minimaalne maksustatav andmehulk on 1024 baiti (=1 kB) (<http://www.emt.ee/est/html4/?smg=teenused&schg=gprs>)

Tabel (Tabel 2) näitab, kui palju aega ja andmehulka kulus selle klipi vaatamiseks erinevate võrguparameetrite korral.

Parameeter	väärtaus	Katkestuseta ettemängimine (aeg, s)			Maksustatav andmehulk (kB)		
		TCP	HTTP	UDP	TCP	HTTP	UDP
Piiramata		2:08	2:10	2:08	304	309	319
Latentsus	500 ms	+	+	+	+	+	+
	1500 ms	2:29	2:40	2:29	291	256	274
	3000 ms	-	-	2:50	-	-	306
Üleslüli latentsus	1500 ms	2:19	2:25	2:19	289	274	292
	3000 ms	2:30	2:35	2:30	270	237	294
Allalüli latentsus	1500 ms	2:20	2:25	2:19	260	298	285
	3000 ms	2:29	2:34	2:30	291	225	284
Läbilaskevõime	40 kb/s	+	+	+	+	+	+
	20 kb/s	-	-	2:13	-	-	244
	12 kb/s	-	-	-	-	-	-
Üleslüli läbilaskevõime	10 kb/s	2:10	2:12	2:10	282	307	320
	3 kb/s	2:14	2:21	2:14	265	328	297
	1 kb/s	2:33	2:43	2:34	295	304	282
Allalüli läbilaskevõime	20 kb/s	2:12	2:14	2:12	233	240	249
	12 kb/s	-	-	-	-	-	-
	7 kb/s	-	-	-	-	-	-
Paketikadu	1%	2:09	2:12	2:09	285	288	234
	10%	2:18	-	2:14	298	-	230
Üleslüli paketikadu	1%	2:09	2:13	2:08	304	309	319
	10%	-	-	2:10	-	-	287
	30%	-	-	-	-	-	-
Allalüli paketikadu	1%	+	+	+	+	+	+
	10%	2:10	2:18	2:11	311	313	233
	30%	-	-	-	-	-	-
Bitivea tõenäosus	1e-9	2:08	2:13	2:09	305	308	317
	1e-6	2:08	2:10	2:08	306	290	286
	1e-5	+	+	+	+	+	+
Üleslüli bitivea tõenäosus	1e-6	2:09	2:13	2:08	305	308	318
	1e-5	2:11	2:13	2:09	305	310	318
Allalüli bitivea tõenäosus	1e-5	2:09	2:10	2:13	344	325	236
GPRS	normGPRS	2:18	2:21	2:18	276	253	275
	badGPRS	-	-	2:33	-	-	240
	bench4	-	-	-	-	-	-

*“- “tähendab, et ettemängimisel oli katkestusi puhvri tühjenemise tõttu.”+
“tähendab, et ettemängimine toimus katkestuseta, aga testitulemuste puuduliku kogumise tõttu pole andmeid. Katsete suure mahu tõttu jäeti osa katseid tegemata.

Tabel 2. Ettemängimiseks kulunud ajad ja edastatud andmehulgad erinevate võrguparameetrite korral

Milliste võrgu parameetrite korral edastuse maksumus on kõige väiksem ajapõhise maksustamisega, milliste võrgu parameetrite korral maksumus on kõige väiksem andmehulga põhise maksustamisega, seda vaatame järgnevast tabelist (2.3.9).

Maksumus (2,49 kr 60 sek (0,0415 kr/s) ; 40 kr 1024 kB (0,0390625 kr/kB)

Parameeter	Väärtus	Ajapõhine			Andmehulga põhine		
		TCP	HTTP	UDP	TCP	HTTP	UDP
Piiramata		5,31	5,40	5,31	11,88	12,07	12,46
Latentsus	1500 ms	6,18	6,64	6,18	11,37	10,00	10,70
	3000 ms			11,21			11,95
Üleslüli latentsus	1500 ms	5,77	6,02	5,77	11,29	10,70	11,41
	3000 ms	6,23	6,43	6,23	10,55	9,26	11,48
Allalüli latentsus	1500 ms	5,81	6,02	5,77	10,16	11,64	11,13
	3000 ms	6,18	6,39	6,23	11,37	8,79	11,09
Läbilaskevõime	20 kb/s			5,52			9,53
Üleslüli läbilaskevõime	10 kb/s	5,40	5,48	5,40	11,02	11,99	12,50
	3 kb/s	5,56	5,85	5,56	10,35	12,81	11,60
	1 kb/s	6,35	6,76	6,39	11,52	11,88	11,02
Allalüli läbilaskevõime	20 kb/s	5,48	5,56	5,48	9,10	9,38	9,73
Paketikadu	1%	5,35	5,48	5,35	11,13	11,25	9,14
	10%	5,73		5,56	11,64		8,98
Üleslüli paketikadu	1%	5,35	5,52	5,31	11,88	12,07	12,46
	10%			5,40			11,21
Allalüli paketikadu	10%	5,40	5,73	5,44	12,15	12,23	9,10
Bitivea tõenäosus	1e-9	5,31	5,52	5,35	11,91	12,03	12,38
	1e-6	5,31	5,40	5,31	11,95	11,33	11,17
Üleslüli bitivea tõenäosus	1e-6	5,35	5,52	5,31	11,91	12,03	12,42
	1e-5	5,44	5,52	5,35	11,91	12,11	12,42
Allalüli bitivea tõenäosus	1e-5	5,35	5,40	5,52	13,44	12,70	9,22
GPRS	normGPRS	5,73	5,85	5,73	10,78	9,88	10,74
	badGPRS			6,35			9,38

Tabel 3.

Tabel 4. Ettemängimise maksumus ajapõhise ja andmehulgapõhise maksustamise korral erinevatel võrgutingimustel

Tabelist (2.3.9) on näha, et seda klippi on odavam vaadata, kui aja eest peab maksma. Erinevus kuni poole võrra. Kuid ajapõhiselt maksustatud side edastuskiirus on 9600 b/s, mis selle multimeediaklipi vaatamiseks ilma katkestusteta on vähe.

Tabelist on näha veel, et aja eest maksustamise korral võrgutingimuste halvenemisel hakkab kasutaja maksma rohkem, aga andmehulga eest maksustamise korral võrgutingimuste halvenemisel hakkab kasutaja maksma vähem. Järelikult kasutaja jaoks õiglasem maksustamise viis on andmehulga põhine.

Kokkuvõte

Töö eesmärk on koostada eestikeelne väike ülevaade sellest, kuidas on tehtud ja võiks teha videokonverentse ja nende salvestusi ning neid internetist (kõige üldisemalt öeldes) kätte saada. Töös on ka toodud Side loengute tegemise kogemused, et järgmised saaks õppida teiste vigadest juba eesti keeles. Lisaks on väike ülevaade analoog heli- ja videosignaali digitaliseerimisest, saadud andmete pakkimisest ja edastamisest. Lühidalt on kirjeldatud tarkvara, mida erinevatel aegadel Side loengutes kasutati nende internetti edastamisel. Proovitud on ka tarkvara, mis lisavad ka ettevalmistatud õppematerjalid (Microsoft PowerPoint slaidid) selle videoga kaasa.

Kõige mugavam on kasutada sellist tarkvara, millega kasutaja saab soovitud loenguvideo võimalikult väikese vaevaga kätte, näiteks 1 hiireklõpsuga veebibrauserist. Uuritud programmidest oli kõige mugavam kasutada Real Networks tarkvara.

Töö teises osas uuriti, kuidas erinevad edastusvõrgu parameetrid igaüks eraldi ja koos mõjutavad mobiilse andmesidevõrgu parameetritega arvestades loodud videoklipi edastamist. Multimeedia andmevoog, mis on kodeeritud edastuskiirusega on 20 kb/s, on mobiilandmesidevõrgu GPRS kaudu kvaliteetselt kuulatav-vaadatav, kui võrk on vähe koormatud.

Võrgukasutuse eest maksustamisel andmehulgapõhine maksustamine on õiglasem kasutaja suhtes kui ajapõhine maksustamine.

Kasutatud kirjandus

1. Adobe Premiere Technical Guides, Video codec compression methods, <http://www.ledet.com/coolstuff/software/premiere/methods.pdf>
2. Avalikud loengud, <http://ernie.itcollege.ee/koostoo/avalikudloengud.php>
3. Berkeley Internet Broadcast System, <http://bmrc.berkeley.edu/bibs/>
4. BIBS: Frequently Asked Questions, <http://bmrc.berkeley.edu/bibs/bibs-faq.html>, Last modified: August 25, 2001
5. BMRC Lecture Browser Demo, <http://bmrc.berkeley.edu/frame/projects/lb/index.html>
6. S.E. Deering, Host extensions for IP multicasting. Aug-01-1989, RFC1112
7. Joe Duran, Charlie Sauer, Mainstream Videoconferencing, A Developer's Guide to Distance Multimedia, Addison Wesley Longman Inc 1997.
8. Encoder FAQ, Microsoft Corporation, <http://www.microsoft.com/windows/windowsmedia/wm7/encoder/faq.asp>
9. Epan, <http://www.et-inf.fho-emden.de/~tobias/epan/>, <http://packages.debian.org/unstable/net/epan.html>
10. W. Fenner, Internet Group Management Protocol, Version 2. November 1997. RFC2236
11. Ross Finlayson, The UDP Multicast Tunneling Protocol, Internet-Draft 2001.03.02, <http://www.ietf.org/internet-drafts/draft-finlayson-umtp-06.txt>
12. Free On-Line Dictionary Of Computing, <http://foldoc.doc.ic.ac.uk/foldoc/>
13. David Ginsburg, ATM solutions for enterprise internetworking, Second edition, Addison Wesley
14. Simon Haykin, Digital Communications, John Wiley & Sons, Inc, 1988, lk 199
15. Internet Multicast Addresses, <http://www.iana.org/assignments/multicast-addresses>
16. MATE Product Sheet, revision c, Ericsson Radio Systems. productsheet.pdf
17. David McDysan QoS & Traffic Management in IP & ATM Networks, McGraw-Hill 2000
18. Microsoft Windows NetMeeting 3 Service Pack 1 Readme, Microsoft Corporation, 1999, C:\Program Files\NetMeeting\NETMEET.HTM
19. Network Text Editor, <http://www-mice.cs.ucl.ac.uk/multimedia/software/nte/>
20. OpenH323. H.323 standards, <http://www.openh323.org/standards.html>
21. J. Postel. Transmission Control Protocol. Sep-01-1981. RFC0793
22. J. Postel, User Datagram Protocol. . Aug-28-1980, RFC0768
23. RealServer Administration Guide RealServer 8.0, Chapter 3: Overview of RealServer, RealNetworks Inc, 2000, Manual/realsrvr.htm
24. Remote Language Teaching, <http://www-mice.cs.ucl.ac.uk/multimedia/software/relate-ui/>
25. H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. Audio-Video Transport Working Group, January 1996. RFC1889
26. H. Schulzrinne, A. Rao, R. Lanphier. Real Time Streaming Protocol (RTSP). April 1998. RFC2326
27. T.J. Socolofsky, C.J. Kale. TCP/IP tutorial. Jan-01-1991. RFC1180
28. TechEncyclopedia, <http://www.techweb.com/encyclopedia/>
29. User Guide for SDR v2.5, University College London, Computer Science Department, <http://www-mice.cs.ucl.ac.uk/multimedia/software/documentation/sdr-userguide.zip>

30. User Guide for RAT v4.2, University College London, Computer Science Department, <http://www-mice.cs.ucl.ac.uk/multimedia/software/documentation/rat4-ug-v4-0011.doc>
31. Userguide for vic v2.8, University College London, Computer Science Department, <http://www-mice.cs.ucl.ac.uk/multimedia/software/documentation/vic-userguide.zip>
32. User Guide for WBD. Version 1, September 1998, University College London, Computer Science Department, <http://www-mice.cs.ucl.ac.uk/multimedia/software/documentation/wbd-userguide.zip>
33. Veebipõhine WebCT õpikeskkond, <http://www.ut.ee/AvatudYlikool/haridustehnoloogia/WebCT.html>
34. Videokonverents, korduma kippuvad küsimused, <http://www.datel.ee/it/side/index.htm?kkk.shtml~datsidesisu>
35. Sirje Virkus, Avatud õppe ja kaugkoolituse mõiste, <http://www.tpu.ee/~i-foorum/ifoorum5/sirjeter.htm>
36. Virtuaalõpe IT kollezhis, <http://www.itcollege.ee/oppimine/virtuaalope.php>
37. Webcast.berkeley:FAQ, <http://webcast.berkeley.edu/faq.html>
38. WebCT, <http://www.webct.com/>
39. Webopedia: Online Dictionary for Computer and Internet Terms, <http://www.webopedia.com/>