

TARTU ÜLIKOOL
MATEMAATIKA-INFORMAATIKATEADUSKOND
Arvutiteaduse instituut
Infotehnoloogia eriala

Madis Kure

**TeX animatsioonide loomine ja nende
presenteerimine**
Bakalaureusetöö (6 EAP)

Juhendaja: lektor Tõnu Tamme

Autor: „juuni 2010
Juhendaja: „juuni 2010

Lubada kaitsmisele
Professor „juuni 2010

TARTU 2010

Sisukord

Sissejuhatus.....	3
1. Graafiliste kolmemõõtmeliste animatsioonide loomine.....	4
1.1 MetaPost.....	4
1.2 PSTricks.....	5
1.3 Pgf/Tikz.....	6
1.4 Vahendid piltanimatsioonide kaadritest terviku kokkupanemiseks.....	7
2. Lihtsamad animatsioonid ja animatsioonide paigutamine presentatsiooni.....	8
2.1 Slides.....	9
2.2 Prosper.....	9
2.3 Beamer.....	9
2.4 TeXPower.....	10
2.5 PPower4.....	11
2.6 Animate.....	11
2.7 Movie15.....	12
2.8 Flashmovie.....	12
3. Presentatsioonivahendid.....	13
3.1 Adobe Acrobat Reader.....	13
3.2 OpenOffice Impress.....	13
3.3 Microsoft Office PowerPoint.....	14
4. Näited.....	15
4.1 Tõeväärtustabel.....	15
4.2 Silinder.....	17
Kokkuvõte.....	18
Creating and presenting TeX animations.....	19
Kasutatud kirjandus.....	20
Lisad.....	22

Sissejuhatus

TeX on süsteem, mis on kirjutatud Donald Knuth poolt ilusate raamatute loomise jaoks (eriti kui seal on palju matemaatikat). LaTeX on populaarne TeX makrode pakett, mis lisab viimasele funktsionaalsust ning teeb kasutamise hõlpsamaks. Animatsiooni all on mõeldud üksteisest vähesel määral erinevate objektide sellist esitlemist, mis jätab vaatajale objekti sujuva liikumise või muutumise mulje. [1]

Selle töö eesmärgiks on anda ülevaade TeX süsteemi ja sellega seotud vahendite animeerimis- ja presenteerimisvõimalustest. Kirjeldatakse asjakohaseid programme ja nende laiendusi, mis võimaldavad luua animatsioone ning tuuakse välja nende eelised/puudused üksteisega võrreldes. Esimene peatükk on pühendatud ruumiliste objektide loomisele. Kuna enamikel juhtudel saame tulemuseks eraldiseisvad kaadrifailid, siis on kirjutatud ka nendest terviku kokkupanemisest. Töös kaetakse Unix ja Windows süsteemis töötavad lahendused, sest kasutatavad vahendid ei pruugi alati olla kõigis operatsioonisüsteemides kättesaadavad. Teine peatükk räägib lihtsamate animatsioonide loomisvõimalustest ning kuidas erinevaid animatsioone presentatsiooni paigutada. See jaotis on kõige mahukam ja põhjalikum kuna katab töö seisukohalt tähtsaima osa. Kolmandas peatükis on juttu populaarsetest esitlusvahenditest, nende võrdlus ja kasulikkus animeerimise koha pealt. Töö lõppu on lisatud kaks näidet koos üksikasjalike kirjeldustega, mille eesmärgiks on lahenduskäikude detailidest parema arusaama loomine. Antud temaatikast on kaetud vähemalt kõige populaarsemad võimalused (autori subjektiivne hinnang vastavalt materjalide rohkusele, kättesaadavusele, levikule) ning nende toimimise kontrollimine ühes kaasaegses Windows või Unix süsteemiga keskkonnas.

Katsete tegemisel ning nende põhjal loodud/otsitud näitematerjalidel on põhirõhk asetetatud eelkõige tõeväärtustabelitele ja kolmemõõtmeliste matemaatilistele objektidele (teist järku pinnad). Tõeväärtustabeli puhul keskendume eelkõige selle esitlusele nii, et tõeväärtusi lisandub tabelisse animatsiooni käigus. Teist järku pindade puhul on kõige huvipakkavam nende tekkimise, liikumise iseloomustamine. Töös on kajastatud ka muud asjakohased vahendid, mis kumbagi võimalust otseselt ei pruugi pakkuda. Samas ei käsitleta siin eeldefineeritud (animatsiooni)efektide lisamist slaididele ja nendel leiduvatele elementidele, vaid lausete, tabelite järk-järgulist ehitamist/täitmist või kolme-mõõtmeliste objektide liikumist, loomist nendest parema ruumilise ülevaate andmiseks.

1. Graafiliste kolmemõõtmeliste animatsioonide loomine

Olenevalt sellest, missuguseid animatsioone me luua soovime, on vaja valida endale sobilik TeX makro. Lihtsamate objektide kirjeldamiseks või joonistamiseks ei pruugi eriti spetsiifilist laiendust vaja minna. Väga palju annab ära teha ka ainult LaTeX võimalusi või piltide jaoks selle laiendusi PSTricks, Pgf/Tikz kasutades. Kui nende võimalustest väheks jääb (näiteks ruumilisuse efekti tekitamiseks), siis saab kasutada muid pakette, millest on antud peatükis ka suur osa kaetud.

Tasub tähele panna, et mitmete vahenditega ei saa luua kohe valmisanimatsioone, vaid erinevates formaatides kaadri faile. Tsüklite ning muutujate kasutamine lihtsustab siin oluliselt nende koostamist. Kaadreid annab vajadusel edasi töödelda teiste TeX laienduste või muude programmidega. Enne tasuks aga kindlaks teha, et kogu protseduur sellega väga laiali ei valguks või keeruliseks ei kujuneks. Samuti tasub siinkohal märkida, et mitmetel levinud pilditötlusporgammidel (nagu näiteks Adobe Photoshop, GIMP) on ka võimalusi animatsioonide loomiseks.

1.1 MetaPost

MetaPost on joonistamiskeel, mis on suures osas tuletatud Donald Knuth'i Metafont keelest. Ta on üpriski populaarne piltide joonistamiseks, mida TeX dokumentidesse lisada tahetakse. MetaPost'ile on erinevate autorite poolt loodud mitmeid laiendusi selle funktsionaalsuse suurendamiseks. Allpool on eraldi välja toodud levinumad neist, mis lisavad võimaluse kirjeldada kolmemõõtmelisi objekte, kuna see vahendil algselt puudub. Tuleks tähele panna, et loetletud laiendustega animatsioonide loomine on autorite poolt tehtud kõige lihtsamaks Unix operatsioonisüsteemiga (näited, skriptifailid). Teisi süsteeme kasutades peab ise rohkem vaeva nägema. [1]

3d/3dgeom

Antud pakett on kõige esimene dokumenteeritud Metapost laiendus, mis käsitleb kolmemõõtmelisust. Ta on alljärgnevatega võrreldes vähem võimalusi pakkuv. Autor on omalt poolt kaasa pannud ka Unix skriptifaili, millega saab loodud kaadri failidest GIF animatsiooni kokku panna. Skript kasutab mitmeid erinevaid vahendeid (awk, ghostscript, pnmcut, ppmquant, ppmtogif, gifmerge), mille olemasolu on vajalik selle toimimiseks. Teiste operatsioonisüsteemide ja failiformaatide jaoks peab skriptifaili muutma või muid vahendeid kasutama. [2]

M3d

M3d on teiste nimetatud Metapost laiendustega võrreldes võimeline produtseerima keerulisemaid ja ilusamaid diagramme. Samuti nagu 3d/3dgeom puhul on parim võimalus GIF animatsiooni koostamiseks kasutada kompileerimisel loodavat skriptifaili, mis peaks töötama UNIX süsteemide peal. Skriptifail kasutab Convert (Imagemagick'uga kaasas) ja Sed programmi. Joonisel 1 on näha ühte näidet selle võimalustest. [3]

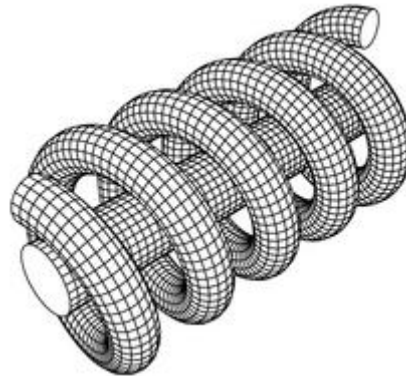
Featpost

Featpost on osaliselt mõeldud füüsikadiagrammide joonistamiseks, kuid pakub ka võimalusi kolmemõõtmeliste objektide ja animatsioonide koostamiseks. Nende loomiseks

kasutatakse Python skripti ning MPEG videote produtseerimiseks on vajalikud Ghostscript ja Netpbm. Kahjuks ei ole infot nende kasutamise kohta dokumenteeritud. [4]

(GNU) 3DLF

Ei ole puhas Metapost pakett, kuna see on kirjutatud C++ keeles, kuid tulemuseks saadakse siiski Metapost'i fail. Autor on loonud mugava vahendi TeX dokumentidesse sisestamiseks mõeldud kolmemõõtmeliste objektide loomiseks. Samuti on võimalik seda kasutada animatsioonide koostamiseks. Negatiivse külje pealt tooks välja selle, et laiendus on mõeldud Unix süsteemil jooksutamiseks ning autori poolt kaasa pandud materjal on raskelt hoomatav. [5]



Joonis 1. M3d laiendusega loodud ruumiline objekt [3].

1.2 PSTricks

PSTricks (PostScript Tricks) on hulk makrosid, mis lubavad PostScripti (programmeerimiskeel) joonistusi otse TeX, LaTeX koodi lisada. PSTricks on populaarne vahend piltide loomiseks TeX dokumentide jaoks. Samuti nagu Metapost puhul ei olnud tal algselt võimalusi kolmemõõtmeliste objektide kirjeldamiseks, aga praeguseks on erinevate autorite poolt loodud mitmeid laiendusi, mis seda lubavad. Levinumad sellised pakettid on eraldi välja toodud allpool. PSTricks pakub ka ülekatmist (*overlays*) võimalust, mis võib huvi pakkuda lihtsamate animatsioonide loomisel, kui töö on vormistatud slaididel. [6]

Pst-3d

Üks vanemaid PSTricks laiendusi, mis võimaldab tekitada ruumilisuse efekti erinevate vaatepunktide, -nurkade kirjeldamisega. Ta toetab ainult paralleelprojektsioone (mitte näiteks sfäärilist vaatenurka). [7]

Pst-vue3d

Selle laienduse üheks eesmärgiks on kolmemõõtmeliste objektide joonistamine nii, et varjatud osad elimineeritakse. Vaatepunkt on kirjeldatud sfääriliste koordinaatidega. Mitmed ruumilised kujundid (näiteks silinder, püramiid, sfäär) on eeldefineeritud, mis võimaldab nende kerget joonistamist. Samuti saab kirjeldada ruumilisi punkte, jooni nii Cartesiuse kui ka sfääriliste koordinaatidega. [8]

Pst-3dplot

Mõeldud kolmemõõtmeliste objektide joonistamiseks. Defineeritud on ruumilised koordinaatteljed, mida saab joonistada erinevate vaatenurkade alt. Telgedele annab märkida ühikuid ning nende jaoks mõeldud punkte/jooni nii, et kogu koordinaatteljestik kompo-

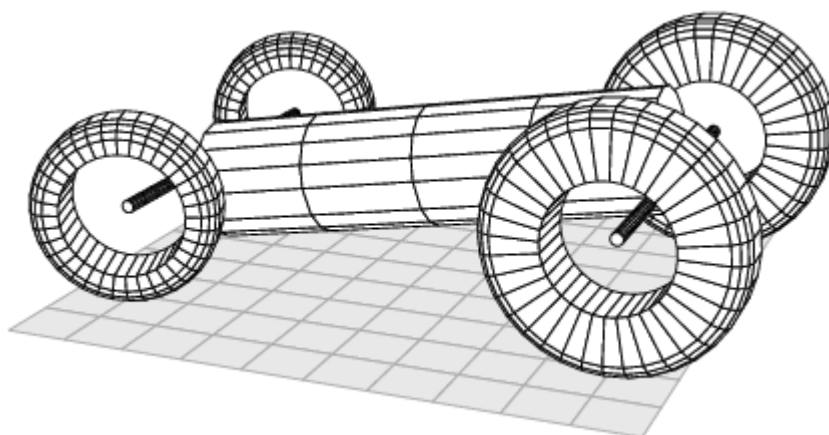
mentidega jätab kolmemõõtmelise efekti. Defineeritud on ka näiteks ruumiline punkt, joon, kolmnurk, risttahukas, sfäär. [9]

Pst-ob3d

See laiendus on mõeldud algeliste kolmemõõtmeliste objektide joonistamiseks. Hetkel on seal defineeritud ainult tavaline kuubik ning täring (punktidega kuubik), mida saab esitada tavaliste või ümarate (mahalõigatud) nurkadega. [10]

Pst-solides3d

Tegu on kõige mahukama ja rohkeid võimalusi pakkuva PStricks laiendusega, mis tegeleb kolmemõõtmelisusega. Vahend võimaldab luua 3D vaateid eeldefineeritud või kasutaja poolt kirjeldatud objektidest. Saab määrata vaate-, valgustuspunkti koordinaate. Laiendus pakub päris kõvasti eeldefineeritud objekte alatest koordinaatteljestikust, punktist, vektorist kuni mitmetahuliste kuubikute, erinevate sfääride ja pindadeni. Erinevalt teistest nimetatud pakettidest pakub Pst-solides3d näiteks veel võimalusi kujundeid painutada, tükke küljest ära võtta, pooleks lõigata, kokku panna. Joonisel 2 on näha selle laienduse abiga kujundatud näidispiilt. [11]



Joonis 2. Pst-solides3d laiendusega loodud ruumiline objekt [12].

1.3 Pgf/Tikz

Pgf on TeX pakett, mis on mõeldud graafiliste objektide joonistamiseks ning pakub omalt poolt ka piltide ülekاتمise võimalust lihtsamate animatsioonide tegemiseks. Erinevalt PStricks laiendusest saab sellega luua otse nii PDF kui ka PostScript väljundit. Antud vahendil paistab hetkel olevat ainult üks laiendus (eraldi välja toodud allpool), mis võimaldab kirjeldada ruumilisi objekte. [13]

Tikz-3dplot

See laiendus pakub võimalusi joonistada kolmemõõtmelisi koordinaatsüsteeme ja lihtsaid ruumilisi diagramme. Saab defineerida enda jaoks sobilikku koordinaatsüsteemi, mille järgi objekte paigutada. Hetkel on olemas vahendid näiteks punktide, joonte, sfääriliste pindade joonistamiseks. [14]

1.4 Vahendid piltanimatsioonide kaadritest terviku kokkupanemiseks

Juhul kui eraldiseisvad kaadrifailid, manuaalne kaadrite vahetus või tulemusefaili(de) formaat ei ole sobilikud/meelepärased, siis siin on nimetatud mõningad eraldiseisvad programmid, millega olukorda parandada annab. Nende mainimise põhjuseks on see, et nad aitavad TeX-s koostatud animatsioonide kaadrifailide kokku panna üheks tervikuks, mida mõningad eelpoolnimetatud vahendid ka ära kasutavad. Samuti pakuvad allpool loetletud programmid tihtipeale omapoolseid võimalusi failiformaadi muutmiseks ning animatsioonikäigu juhtimiseks, reguleerimiseks (näiteks kui kiiresti kaadreid vahetatakse). Enne siit abi otsimist tasub tutvuda järgmise peatüki Animate paketiga, mis võimaldab samalaadseid toiminguid.

Netpbm

Tegu on tööriistade kogumikuga, mis on mõeldud graafikafailide manipuleerimiseks ja peaks töötama kõigi enamlevinud operatsioonisüsteemidega (Unix, Windows, Mac). Programmil on mitmeid võimalusi alates failiformaatide muutmise, pildi vähendamise kuni animatsioonide kokkupanekuni. Negatiivseks küljeks võib lugeda graafilise kasutaja-liidese puudumist. Mõningate eelpool nimetatud Metapost laiendustega on autor kaasa pannud skriptifaili animatsioonide tekitamiseks, mis kasutab selle vahendi võimalusi. [15]

Gifmerge

Vahendi peamine eesmärk (nagu nimest võib välja lugeda) on GIF formaadis pildifailidest GIF animatsiooni kokkupanek. Võimalik määrata ka näiteks läbipaistvust, kaadrite vahetuse kiirust ning mitu korda animatsiooni läbitakse. Antud programmi on kasutatud 3d/3dgeom laienduse animatsiooni koostamise skriptifailis. [16]

SWFTools

SWFTools on vahendite kogumik, mis on mõeldud töö tegemiseks Adobe Flash (SWF) failidega. Võimalik tekitada muuseas ka PDF, JPEG, PNG, GIF (ka animeeritud) failidest SWF filmi, et neid näiteks Adobe Flash Player'iga esitleda. Pakub ka muud kasulikku funktsionaalsust nagu SWF failide lugemine, kombineerimine ning peaks töötama enamlevinud operatsioonisüsteemidega. [17]

ImageMagick

Tegu on programmiga, mis on mõeldud piltide loomiseks ning nendega manipuleerimiseks. ImageMagick toetab suurt hulka failiformaate nagu näiteks GIF, JPEG, PDF, PNG, PostScript ning lubab faile ühest formaadist teise konvertida. Vahendiga on võimalik pilte ümber pöörata, peegeldada, neile efekte lisada, neist GIF animatsiooni kokku panna ja palju muud. Selle vahendi võimalusi on kasutatud ka Metapost M3D laienduse animatsiooniskriptis. [18]

2. Lihtsamad animatsioonid ja animatsioonide paigutamine presentatsiooni

Antud peatükis antakse ülevaade TeX vahenditest, millega saab lihtsamaid animatsioone (tabeleid, tekste, loetelusid ja muud mittepildilist) luua. Samuti kirjeldatakse võimalusi, kuidas nii otse TeX dokumendis kirjeldatud kaadri faile kui ka eelnevalt valmis tehtud animatsioone PDF slaidile paigutada. Tabel 1 annab ülevaate käsitletud laiendustest ja nende asjakohasest funktsionaalsusest ning tabel 2 iseloomustab mõningates näidetes kasutatavat tõeväärtustabelit.

Tabel 1. Animeerimisvõimalusi pakkuvad TeX laiendused.

	Animeerimis -võimalused	Animatsiooni kaadrite automaatne läbimine	Kontroll- nuppude/ linkide lisamise võimalus	Kaadrifaili- dest üheslaidilise animatsiooni kokkupanek	Eraldiseisva animatsiooni siselugemine
Slides	v				
Prosper	v				
Beamer	v	v			
TeXPower	v	v	v		
PPower4	v		v		
Animate	v	v	v	v	v
Movie15		v	v		v
Flashmovie		v	v		v

Tähele peaks panema seda, et Slides, Prosper, Beamer on dokumendiklassid (*documentclass*), mis tähendab, et neist on korruga võimalik kasutada ainult ühte. Koos dokumendiklassi defineerimisega võib *usepackage* käsuga sisse lugeda suvalise arvu pakette, mistõttu võib öelda, et TeXPowder, PPower4, Animate, Movie15, Flashmovie saab vähemalt teoreetiliselt (kombineerimisel võivad tekkida probleemid ühilduvusega) kõiki korruga kasutada. Slides, Prosper, Beamer, TeXPowder, PPower4 on mõeldud suures osas esitluse slaidide loomiseks – pakuvad mitmeid võimalusi nende koostamiseks ning ilmestamiseks. Ülejäänud tabelis 1 nimetatud laiendused on loodud just animatsioonide koostamiseks või selle tulemusefaali sisestamiseks.

Tabel 2. Tõeväärtustabel.

Tõeväärtustabel			
p	q	p ja q	p või q
T	T	T	T
T	V	V	T
V	T		
V	V		

2.1 Slides

See klass asendab vana SLiTeX programmi ja on mõeldud eelkõige tulemuse printimiseks läbipaistavale filmile/kilele, et seda projektoriga näidata. Kui tulemust tahetakse digitaalsena/PDF formaadis arvuti monitoril kuvamiseks või otse arvutist projekteerimiseks, siis soovitatakse kasutada Prosper või mõnda muud klassi.

Slides klassil on animeerimise seisukohalt defineeritud kaks kasulikku käsku: *invisible* ja *visible*, mis (nagu nende inglisekeelsest nimetusest välja võib lugeda) muudavad elemente vastavalt nähtamatuks ja nähtavaks. Selle omadusega on võimalik näiteks tekstis mõningad sõnad nähtamatuks muuta ja järgnevatel slaididel järjest nähtavaks teha vastavalt sellele, kuidas parasjagu soovitakse. Tähele peaks siinkohal panema seda, et vahepeal piisab ka lihtsalt elementide mittekirjeldamisest selle asemel, et neid *invisible* käsuga varjata. Näiteks kasutatud tõeväärtustabelis (Tabel 2) annab elemendi kirjeldamata jätmise sama tulemuse kui *invisible* kasutamine. Mõnikord tekivad aga probleemid tühikute või moonutustega, mis elementide ärajätmisel tekivad, kus võivad nimetatud käsud kõvasti peavalu säästa.

Eelpool nimetatud võimalust kasutades saab tekitada animatsiooni efekti, kuid kaadrid tulevad kõik eraldi slaidile ja neid tuleb manuaalselt vahetada. Seetõttu on meetod sobilikum pigem lühemate animatsioonide jaoks, kus vajalikud slaididevahelised pausid on suhteliselt pikad, nende kestvus varieerub ning pole täpselt ette teada. [19]

2.2 Prosper

Prosper on LaTeX klass, mis on mõeldud kõrgekvaliteediliste slaidide koostamiseks. Sellega loodud faile saab lõpptulemusena kätte kahes formaadis: PS ja PDF. Prosper pakub võimalust ka animatsioonide loomiseks ülekatmistele (*overlays*) tulemusena. Paigutades slaidid *overlays* makro sisse on võimalik informatsiooni slaidi peal näidata etappide kaupa elemente lisades/asendades animatsiooniefekti tekitamiseks. Tasub märkida, et vajaminevaid makrosid on mõttekas kasutada ainult *pdf* režiimis - *ps* režiimis nad töötavad valesti või üldse mitte.

Samuti nagu Slides klassi puhul tekivad kaadrid eraldi slaididele ja neid tuleb ise manuaalselt vahetada. Järelikult on prosper klassi pakutud võimalus sobilik pigem lühemateks animatsioonideks, kus animatsiooni tahetakse läbida aeglase tempoga ise kaadritevahelise pausi pikkust valides. Tähele tasuks panna ka kompileerimisel tekkivaid hoiatusi, mis käsevad vahepeal uuesti kompileerida dokumendi õigeks vormistamiseks. [20]

2.3 Beamer

Beamer on rohkete võimalustega LaTeX laiendus presentatsioonide loomiseks, mis pakub omalt poolt ka funktsionaalsust animatsioonide koostamiseks. Väljundina on võimalik saada nii PS kui ka PDF faili.

Defineeritud on käsk *pause*, millega saab infot edastada slaididele järk järgult objekte lisades eelmiseid kustutamata (ei ole vaja korduvalt osi dokumenti luues iga slaidi jaoks eraldi korrata). Tulemus on sama nagu oleks tehtud mitu slaidi, kus esimesel on üks osa

esitusest ja järgnevatel korratakse eelmiste sisu ning lisatakse midagi juurde. Eeliseks on ruumi-, töökokkuvaid ning vähem probleeme paigutusega. Vaatajatele aitab see jätta ka muljet nagu lisataks elemente samale slaidile. Juhul kui elemente soovitakse presenteerida nii, et informatsioon tuleb juurde eelneva keskele või ette (mitte järele), siis selleks on olemas ülekatmiste (*overlay*) spetsifikatsioon. Nimetatud võimalusega saab märkida elementidele, mis slaididel nad esinevad. Kasutada annab seda nii, et varjatud element võtab enda alla ruumi, või selliselt, et ta seda ei võta. Kui soovitakse esialgset informatsiooni järgnevatel slaididel asendada, siis selleks on defineeritud *overprint* keskkond. Võimalik on animeerida ka pilte kasutades *includegraphics* ning *multiinclude* (vaja Xmpmulti paketti) käsk, et graafikafaile sisse lugeda. Beamer toetab mitmeid formaate, mille hulgas on ka PDF, PNG, JPG ning Metapost'i loodud graafikafailid.

Beamer pakub animeerimise koha pealt rohkem võimalusi kui Slides ja Prosper klassid, kuid tulemus realiseeritakse samamoodi eraldi slaididele, mis võib olla pikemate animatsioonide korral üpriski ebamugav. Üheks oluliseks plussiks siinkohal on Beameri *animate* käsk, mis võimaldab animatsioonikaadreid automaatselt läbida. Käsu töötamiseks peab kasutama PDF failide puhul aga täisekraani režiim (*full screen mode*). Selle saavutamiseks Adobe Readeriga tuleb valida menüüst *View* → *Full Screen Mode*. [21]

2.4 TeXPower

TeXPower on stiili- ja klassifailide kogumik dünaamiliste onlain (*online*) presentatsioonide koostamiseks LaTeX'ga. Väljund on PDF formaadis ja mõeldud esitlemiseks kasutades Adobe Acrobat Reader'it.

Sarnaselt Beamer'iga on siin defineeritud *pause* käsk, mis töötab samamoodi: tekitatakse objektide järk järgulise lisamise efekt nii, et esimesele järgnevatel slaididel on korratud vana ja lisatud midagi uut. TeXPower'il on veel võimalus kasutada *stepwise* käsku, mida võib võtta kui *pause* edasiarendust: töötab suures osas samamoodi, aga pakub rohkemaid võimalusi. *Stepwise* keskkonnas on võimalik kasutada veel mitmeid erinevaid käsk, mille seast meile huvipakkumad on *overlays* ja *movie*. *Overlays* sarnaneb *pause* käsuga selle erinevusega, et uued elemendid paigutatakse vanade peale. Vanu elemente uute lisamisel alt ära ei kustuta, mistõttu tasub kontrollida, et lõpptulemus ikka soovitu vastab. Antud võimaluse testimisel tekkis kompileerimisel palju vigu, kuid funktsionaalsus ikkagi töötas. *Movie* võimaldab animatsioonislaidide automaatset läbimist mingi kindla eeldefineeritud aja tagant. *Movie* sarnaselt Beamer'i *animate* käsuga töötab ainult täisekraani režiimis. TeXPower pakub veel võimalust esitluses PDF lehtede automaatset vahetamist (Hyperref paketi abil) defineerides *pageDuration{<aeg>}*, kus aeg on sekundites. Kasutades *pageDuration{}* või *stopAdvancing* saab määrata kohta kus see lõppema peaks. Viimaseks huvipakkuvaks osaks on võimalus kasutada navigeerimisnuppe slaididel edasi/tagasi liikumiseks, mis võib kasulik olla eriti pikemate animatsioonide puhul täisekraani režiimis.

Kogu selle kasuliku ja huvitava funktsionaalsuse juures on manuaal kahjuks napisõnaline käskude lahtiseletamisel. Samuti pole toodud näiteid, mis võib kasutamise algajale raskeks teha. Olukorda parandavad natukene vahendi kodulehele üles riputatud erinevad näidistööd. [22-23]

2.5 PPower4

Tegemist on Java programmiga TeX presentatsioonidele dünaamiliste efektide lisamiseks, mille kasutamiseks peab olema arvutil ka Java installeeritud. Selle vahendiga tehakse kõigepealt valmis esialgne PDF fail (kasutades vajalikke stiilifaile) ja peale seda peab eraldi skriptifaili jooksutades laskma PPower4'jal dünaamilised efektid lisada (tekitatakse uus PDF fail). Kõik kasutamiseks vajaliku leiab vahendi koduleheküljelt [24], kuid tuleks tähele panna, et skriptifaili on enda süsteemi jaoks vaja eraldi kohendada.

Samuti nagu Beamer ja TeXPower puhul on siin defineeritud *pause*, millega saab samamoodi elementide slaidile lisamise efekti jätta. Käsk *pauselevel* lubab pausidele lisada järjekorranumbri, mis võimaldab elemente lisada läbisegi (mitte ainult järjest ülevalt alla või vasakult paremale). Näiteks annab tabelitel kõigepealt nähtavaks teha viimase rea ja siis alles eelviimase. *Pauselevel* võimaldab veel vanu elemente uutega asendada ning märkida, mis objekt millistel animatsioonijärkudel on parasjagu nähtav/nähtamatu. Kasutades *multiinclude*, *includegraphics* saame sisse lugeda pildifaile, mida annab animatsiooni tekitamiseks kasutada. PPower4 ei paku võimalust automaatselt slide läbida, aga tal on defineeritud lingid millega saab näiteks animatsiooni algusesse või lõppu liikuda. [24-25]

2.6 Animate

Tegu on LaTeX laiendusega, mis võimaldab erinevatest kaadrifailidest üheleheküljelise PDF animatsiooni luua. TeX'iga seotud vahenditest sobivad näiteks LaTeX-picture, PSTricks, pgf/TikZ poolt genereeritud pildid. Erinevalt Movie15 paketist võimaldab see vahend animeerida vektorgraafikat. Tulemus sarnaneb SWF formaadis animatsiooniga, kuid võtab enda alla rohkem mälu.

Animate pakett pakub käsku *animategraphics* ja keskkonda *animateinline*. Esimene neist võimaldab animatsiooni kokku panna eksisteerivatest kaadrifailidest (sealhulgas ka mitmeleheküljelisest PDF failist, kus iga lehekülg on üks kaader). Selle töötamiseks peab enamikel kaadrifailidel olema mingi ühine osa nimedes, mille järgi neid ära tuntakse ja sisse loetakse. Samuti on tihti vaja, et neil oleksid nimes numbrid, mille järgi neid järjestada. Võimalus on veel animatsiooni automaatne uuesti käivitumine, kui ta lõppu jõuab, ja vajadusel saab ka teha nii, et animatsioon liigub uude kaadrisse ainult hiireklikiga. *Animategraphics* sisestab kaadrifailid tulemusfaili, mis tähendab, et eraldiseisvaid faile tulemusfailis animatsiooni vaatamiseks vaja ei lähe. Selle funktsionaalsuse kohta leiab ka näite peatükis 4.2.

Animateinline koostab animatsiooni objektidest mis ta sisse jäävad (uue kaadri algust märgitakse *newframe* käsuga). Need objektid võivad olla lihtsalt tekstilised või ka LaTeX-picture, PSTricks, pgf/TikZ keskkonnas kirjeldatud pildid. *Animateinline* keskkonnas on võimalik määrata pausikohad, kus animatsioon peatub. Animatsiooni alustamiseks ja uuesti käivitamiseks peale pausi on vaja sellele tulemusfailis hiirega klikata. *Animateinline* keskkonna käsk *multiframe* võimaldab defineerida kolme tüüpi (täisarvud, reaalarvud, dimensioonid või LaTeX pikkused) muutujaid, mis mingi objekti animatsiooni käigus muutuvad. *Animateinline* kasutamise kohta on ka üks näide peatükis 4.1.

Nii *animateinline* kui ka *animategraphics* puhul saab määrata kaadrite läbimise kiirust ning on võimalik lisada nuppe animatsiooni käigu reguleerimiseks. Need nupud pakuvad järgnevaid võimalusi: 1) peatu ja mine esimesele kaadrile 2) samm tagasi 3) mängi

tagurpidi 4) mängi edaspidi 5) samm edasi 6) peatu ja mine viimasele kaadrile 7) vähenda kiirust 8) originaalkiirus 9) suurenda kiirust. Kui animatsioon mängib, siis töölepanekunupud asenduvad pausi nupuga.

Animate on pakett suure hulga kasulike võimalustega, mille kasutamist tasub animatsioonide loomisel LaTeX vahenditega kindlasti kaaluda. Vahendi suureks plussiks eelpool nimetatutega võrreldes on see, et nii *animateinline* kui ka *animategraphics* paigutavad animatsiooni ühele lehele/slaidile. [26]

2.7 Movie15

See laiendus võimaldab PDF dokumentidesse videosid, heli ja 3D objekte sisestada. Meediafailid sisestatakse täielikult PDF väljundisse, mistõttu pole neid peale sisestamist otseselt enam vaja. Movie15 toetab mitmeid erinevaid failiformaate nagu näiteks SWF, AVI, WAV. Võimalik ka GIF animatsioone tulemusfaili sisestada, aga kirjutamise hetkel polnud Adobe Reader võimeline neid mängima.

Meediafailide sisselugemiseks kasutatakse *includemovie* käsku. Võimalik on määrata erinevaid lisavalikuid nagu näiteks *controls*, mis näitab meediat mängivale vahendile kohaseid kontrollnuppe, kui need on saadaval. Samuti saab lasta meediafaili mängida väljaspool PDF faili (*externalviewer*), määrata mängimise kiirust (*rate*) ja nii mõndagi muud. Huvitav funktsionaalsus on veel kolmedimensionaalsete graafikaobjektide sisestamise võimalus PDF faili nii, et kasutaja saab neid interaktiivselt manipuleerida (Movie15 manuaalis [27] lehekülj 16 saab vaadata näidet). Laiendusel on ka võimalus lisada linke (*movieref*) samas dokumendis asuvalle meediafailile.

Katsetamisel SWF failiga tuli määrata filmi *width* (laius) ja *height* (kõrgus), et töötavat tulemust saada. Tihti peale on vaja faili mitu korda kompileerida, et viited õiged saaksid (annab välja hoiatusi, mis seda nõuavad). Tulemusfailis pole pilti enne näha kui selle peal klikata. Adobe Reader küsib ka kas dokumenti usaldada enne kui animatsiooni mängida laseb. [27]

2.8 Flashmovie

See pakett võimaldab *flash* videote (SWF, MP4, FLV jm. laiendid) sisestamist PDF failidesse. Tegu on üprisriki värske laiendusega ja ta töötab ainult Adobe Reader 9 või uuema versiooniga. Põhjuseks see, et Adobe Reader 9! on sisse ehitatud Adobe Flash mängija (vanematel versioonidel mitte), mida antud laiendus ära kasutab. CTAN [28] lehelt leiab ühe näite ja vajaliku stiilifaili (*flashmovie.sty*), kuhu on kirja pandud ka autoripoolne info paketi kasutamisest ning võimalustest.

Vahend lubab alati määrata video jaoks laiust ja kõrgust. Kasutades mõnda välist suurema funktsionaalsusega *flash* mängijat on võimalik lisada näiteks kontrollnuppe. Samuti on seda vaja muude *flash* videote formaatide toetamiseks peale SWF. [28]

3. Presentatsioonivahendid

Selles osas on juttu populaarsematest programmidest, mida tihtipeale presenteerimiseks, esitlemiseks kasutatakse. Meile pakuvad vahendid huvi ka seetõttu, et nad on võimelised mitmeid koostatud animatsioone mängima. Lisaks pakuvad nad omapoolseid võimalusi animeerimiseks või muud kasulikku funktsionaalsust.

Presentatsioonivahendeid on päris mitmeid, neid kõiki üles otsitud ega mainitud pole. Lähemalt on juttu kolmest populaarsest vahendist. Enne nende juurde asumist märgiks veel, et juhul kui GIF, SWF animatsiooni on mingil põhjusel soov näidata muust esitlusest eraldi või muud esitlust ei olegi, siis seda saab üpriski lihtsalt teha mõnda tänapäevast veebibrauserit kasutades. SWF filmi mängimiseks peab installeeritud olema Adobe Flash Player (brauseri plugin), mis on saadaval tasuta. Eksisteerib ka eraldiseisev Flash Player, millega annab neid otse arvutis mängida.

3.1 Adobe Acrobat Reader

TeX esitlused tehakse peamiselt just PDF formaadis ning ka muidu on selles formaadis failid laialdaselt levinud. Acrobat Reader on suure tõenäosusega parim vahend PDF failide esitlemiseks/vaatamiseks ning positiivseks küljeks ka see, et ta on tasuta. Uuemad versioonid võimaldavad samuti erinevates formaatides animatsioonide mängimist. Üheks erandiks siinkohal on hetkel GIF animatsioonid, mis aga ei ole TeX ja selle laienduste võimalusi kasutades otseselt probleemiks.

Adobe pakub ka tasuta edasiarendusi, mis on tunduvalt rohkemate võimalustega kui Reader. Nendega on võimalik näiteks MS PowerPoint faili konvertimine PDF formaati; SWF animatsiooni sisestamine PDF faili ilma TeX kaasabit; erinevate videoformaatide konvertimine FLV formaati, et neid PDF failis mängida saaks. Praeguse seisuga on mainitud vahenditeks Acrobat 9 Standard, Acrobat 9 Pro, Acrobat 9 Pro Extended. Nimetatud vahendite võimalustega lähemalt tutvumiseks ja omavaheliseks võrdluseks tasub uurida Adobe kodulehekülge. [29]

3.2 OpenOffice Impress

Tasuta vahend presentatsioonide loomiseks, mis on alternatiiviks erinevatele LaTeX võimalustele, kus lõpptulemuseks on peamiselt PDF fail. Selle töö kontekstis on ta kasulik, kuna toetab mitmete teiste heli-, videofailide seas GIF, *flash* animatsioone, mida on kerge slaididele lisada. Samuti annab slaidi kiirelt ning automaatselt vahetuma panna ning loetelusid animeerida (järg-järgult elemente lisades). Negatiivseks küljeks on see, et objekte ei anna väiksemate tükide kaupa animeerida (näiteks tabelisse elementide või lausetesse sõnade lisamine). Silma paistab Impress võimalusega kaadrifailidest piltanimatsiooni luua ning see ühe slaidi peale paigutada. Määrata saab kaadrite vahetuse kiirust ja mitu korda animatsiooni läbitakse (võib ka lihtsalt mängima jätta). [30]

3.3 Microsoft Office PowerPoint

Sarnaneb OpenOffice Impress'iga, kuid on tasuline (kirjutamise hetkel oli 2007 versiooni võimalik 60 päeva tasuta proovida, 2010 beetaversiooni sai ka testida). Samuti nagu Impress toetab PowerPoint mitmete heli-, videofailide formaate, sealhulgas GIF ning *flash* faile. GIF animatsiooni slaidile paigutamisel lisanduvad automaatselt kontroll-nupud (*play, pause, move back, move forward*). PowerPoint ei paku kahjuks võimalust manuaalseid animatsioone ühe slaidi peale luua. Sarnaselt OpenOffice Impress'iga võimaldab ta slide automaatselt vahetuma panna, loetelusid animeerida ning ei võimalda väiksemaid objektide osasid animeerida. Silma paistis PowerPoint suure arvu eeldefineeritud animatsiooniefektidega. [31]

4. Näited

Siin peatükis on toodud paar näidet, kuidas mõningate eelpool nimetatud vahenditega animatsioone luua saab. Eesmärgiks on tööprotsessist detailse ülevaate andmine, et lugejal sellest parem arusaam tekiks. Näidetes kasutatakse tõeväärtustabelit ja silindrit, mis on antud töö seisukohalt ka ühed huvipakkuvamad objektid. Täispikkade realisatsioonide koodi- ning tulemusfailide viited leiab lisade alt.

4.1 Tõeväärtustabel

Selle näite puhul on kasutatud *animate* laienduse võimalusi loomaks tõeväärtustabeli animatsioon PDF faili ühele lehele. PDF lehel animatsiooni alustamiseks ning edasiminekuks peale pausi on vaja selle peal hiirega klikata.

Kaader 1 kood

```
\begin{animateinline}{2}
\begin{tabular}{|c|c|c|c|}
\hline
\multicolumn{4}{c}{Tõeväärtustabel}
\hline
p & q & p ja q & p või q
\hline
T & T & T & T
\hline
T & V & V & T
\hline
V & T & & 
\hline
V & V & & 
\hline
\end{tabular}
\end{animateinline}
```

Selgitus

Animatsiooni alustatakse *begin{animateinline}* käsuga. Taga sulgudes olev 2 märgib kaadrite läbimise kiirust (2 kaadrit sekundis). Edasi on kirjeldatud tõeväärtustabel, mida antud kaadril näha soovitakse.

Kaader 1 PDF lehel

Tõeväärtustabel			
p	q	p ja q	p või q
T	T	T	T
T	V	V	T
V	T		
V	V		

Kaader 2 kood

```
\newframe*[5]
\begin{tabular}{|c|c|c|c|}
\hline
\multicolumn{4}{Tõeväärtustabel}\
\hline
p & q & p ja q & p või q\
\hline
T & T & T & T\
\hline
T & V & V & T\
\hline
V & T & V & T\
\hline
V & V & & \
\hline
\end{tabular}
```

Selgitus

Peale esimest kaadrit märgitakse iga uue kaadri algust `\newframe` käsuga. Tärn selle taga tähistab pausi kohta – animatsioon jääb seisma. Kandilistes sulgudes olev 5 märgib animatsiooni läbimise kiiruse muutust kahelt kaadrit viiele kaadrile sekundis.

Kaader 3 kood

```
\newframe
\begin{tabular}{|c|c|c|c|}
\hline
\multicolumn{4}{Tõeväärtustabel}\
\hline
p & q & p ja q & p või q\
\hline
T & T & T & T\
\hline
T & V & V & T\
\hline
V & T & V & T\
\hline
V & V & V & V\
\hline
\end{tabular}
\end{animateinline}
```

Selgitus

Uus kaader algab jällegi `newframe` käsuga. Kogu animatsiooni lõppu tähistab `end{animateinline}`.

Kaader 2 PDF lehel

Tõeväärtustabel			
p	q	p ja q	p või q
T	T	T	T
T	V	V	T
V	T	V	T
V	V		

Kaader 3 PDF lehel

Tõeväärtustabel			
p	q	p ja q	p või q
T	T	T	T
T	V	V	T
V	T	V	T
V	V	V	V

4.2 Silinder

Silindri loomiseks on kasutatud PSTricks ning Pst-solides3d laiendust. Kõigepealt luuakse kaadrifailid (iga kaader eraldi PDF lehel) ning siis kasutades *animate* paketi võimalusi pannakse kokku terviklik animatsioon.

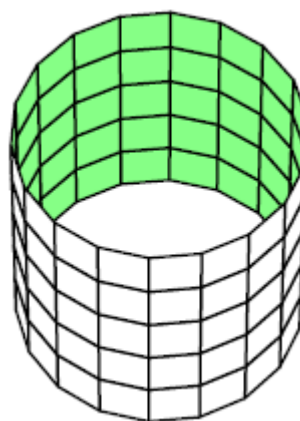
Kaadrite loomise kood

```
\multido{\nz=190+5}{34}{  
\begin{center}  
\begin{pspicture}(-2,-5)(0,-2)  
\psSolid[  
object=cylindrecreux,  
h=1,r=0.5,  
RotX={\nz},  
RotY={\nz},  
fillcolor=white,  
mode=4,  
incolor=green!50]  
(0,0,-3)  
\end{pspicture}  
\end{center}  
\newpage  
}
```

Selgitus

Kasutades *multido* käsku tsükli tekitamiseks luuakse 34 kaadrit. Iga tsükli käigu peal muudetakse muutujat *nz*, millega liigutatakse silindrit x (*RotX*) ning y (*RotY*) telje suhtes.

Esimene kaader PDF lehel



Animatsiooni kokkupaneku kood

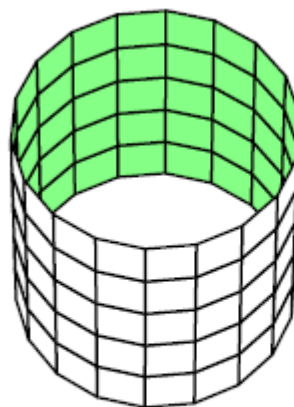
```
\animategraphics[trim=250 500 180  
150,controls]{4}{Lisa2-kaadrid}{}{}  

```

Selgitus

Käsk *animategraphics* abiga on sisse loetud kaadrid failist *Lisa2-kaadrid.pdf*. *Trim* lõikab maha üleliigse valge ruumi pildi ümbert ning *controls* lisab kontrollnupud. Sulgudes olev 4 märgib mitu kaadrit sekundis animatsiooni käigus läbitakse.

Tulemus PDF lehel



Kokkuvõte

Antud töö on kirjutatud eesmärgiga anda ülevaade animatsioonide koostamisest ja nende presenteerimisest TeX vahenditega. Välja on toodud mitmeid erinevaid programme, nende võimalusi ning neid omavahel võrreldud. Suurem rõhk on eelkõige tabelitel ja kolmemõõtmelistel graafilistel objektidel, mis olid vahendite proovimisel peamiseks katseobjektideks.

Alguses on antud ülevaade ning kirjeldatud lühidalt kolmemõõtmeliste objektide loomise võimalusi TeX vahenditega. Kuna mõningatel juhtudel kasutatakse eraldiseisvaid programme animatsiooni kaadritest terviku kokkupanemisel või peab seda vajadusel ise tegema, siis on üks osa pühendatud ka programmidele mis seda võimaldavad. Teine peatükk on töö kõige tähtsam osa, mis räägib lihtsamate animatsioonide koostamisest ning animatsioonide paigutamist presentatsioonide sisse. Need kaks tegevust on võetud kokku ühte peatükki sellepärast, et tihtipeale ühed ja samad paketid pakuvad võimalusi mõlemaks. Juhul kui nimetatud tegevusteks on kasutatud erinevaid vahendeid, siis annab neid üldiselt koos kasutada, mis on ka mõttekas presentatsiooni loomise seisukohalt. Peatüki sissejuhatavas osas on toodud tabel, mis loetletud programmide huvipakkuvatest võimalustest hea ülevaate annab. Enamik võimalusi erinevate vahenditega on töö autor läbi proovinud ning lahatakse ka mõningaid tekkinud probleeme ja nende lahendusi. Kolmandas peatükis on kirjeldatud populaarseid presentatsioonivahendeid, millega annab animatsioone näidata. Võrdluseks on välja toodud ka antud vahendite omapoolsed võimalused animeerimiseks, kui neid esineb. Kõige viimases peatükis on üksikasjalikult lahatud tõeväärtustabeli ning silindri animatsiooni loomisprotsessi. Samuti käsitletakse nende paigutamist esitlusse.

Töoga antakse ülevaatlik pilt mitmete TeX vahenditega animatsioonide loomisest ning võimalustest neid esitlustesse paigutada. Leiti pakette, mis lubavad erinevaid animatsioone koostada/paigutada ühe slaidi peale ning võimaldavad lisada animatsiooni käiku reguleerivat funktsionaalsust (näiteks pausid, kontrollnupud, automaatne kaadrite läbimine eeldefineeritud aja tagant). Oluline osa oli ka tõeväärtustabeli ning silindri (üks võimalik 3D kujund) animatsioonide loomise läbitegemine ja protsessi lahkamine, et nende põhjal potentsiaalsetel lugejatel oleks kergem endale vajalikku luua.

Töö edasiarenduseks sobiks väga hästi kolmemõõtmeliste objektide loomise võimaluste põhjalikum uurimine ning läbivaatus. Samuti tuleks tähele panna, et nii mõnigi pakett on värske – loodud 2009-2010 aastal. Arvatavasti täiendatakse osasid neist (see võib kehtida ka vanemate laienduste puhul, kuid ilmselt väiksema tõenäosusega) ning loota on ka värskete laienduste loomist nii uutelt kui vanadelt autoritelt.

Creating and presenting TeX animations

Bachelor's thesis (6 EAP)

Madis Kure

Abstract

This work has been created with the aim to give an overview of the possibilities TeX offers in creating animations and placing them inside presentation documents (most commonly in PDF format). It is aimed mostly at table and three dimensional object animations. Various possibilities of different TeX packages have been described in detail and their positive/negative sides were brought out. Note that this work is not about adding special animation effects, but about animating object creation and looking at 3D elements from different angles.

The first chapter is about the creation of 3D object animations to give a better overview of them. Different possibilities which TeX offers have been listed with a short description of the range of possibilities they give. Last part of the chapter is about different external programs (they are not directly associated with TeX) which are used and often necessary in putting together animations from the created frames. The second chapter is the most important one which talks about the possibilities of animating simple objects (tables, text for example) and putting different kinds of animations inside your presentation (PostScript and mostly PDF output are supported). Various LaTeX (most common TeX extension for creating documents/presentations) packages have been listed here with what interesting and useful they have to offer within the boundaries of this document. The third part of this thesis gives a short introduction to programs most commonly used in creating presentations. It is about why they are interesting in this context and what kind of animating possibilities they offer. In the last part two examples (one table and a cylinder) have been described. Both the creation of the frames and putting the animation together as a whole into the document are covered in detail.

One of the main achievements of this work is finding and describing different TeX related programs, packages which allow the creation of animations and including them in presentations. The task of finding a way to animate tables by adding elements inside it was completed successfully. Possibilities to animate 3D object with the goal of giving a better overview of them were found as well. Detailed examples of both are included in this document in order to help the readers create similar movies.

Kasutatud kirjandus

1. TeX UK FAQ. <http://www.tex.ac.uk/cgi-bin/texfaq2html?introduction=yes> – viimati vaadatud 29.05.2010.
2. Denis Roegel. Creating 3D animatsions with MetaPost, 1997. <http://ftp.linux.ee/pub/tex-archive/graphics/metapost/contrib/macros/3d/doc/tb57roeg.pdf> – viimati vaadatud 29.05.2010.
3. Anthony Phan. M3D package manual, 2009. <http://www-math.univ-poitiers.fr/~phan/downloads/metapost/m3Dmanual.pdf> – viimati vaadatud 29.05.2010.
4. L. N. Goncalves. Featpost and review of 3D MetaPost packages, 2004. <http://ftp.linux.ee/pub/tex-archive/graphics/metapost/contrib/macros/featpost/doc/tug2004.pdf> – viimati vaadatud 29.05.2010.
5. Laurence D. Finston. 3DLDF User and Reference Manual, 2007. http://www.gnu.org/software/3dldf/manual/user_ref/3DLDF.html – viimati vaadatud 29.05.2010.
6. Timothy Van Zandt. PStricks user's guide, 2003. <http://ftp.linux.ee/pub/tex-archive/graphics/pstricks/base/doc/pstricks-doc.pdf> – viimati vaadatud 30.05.2010.
7. Herbert Voß. Pst-3d documentation, 2010. <http://ftp.linux.ee/pub/tex-archive/graphics/pstricks/contrib/pst-3d/pst-3d-doc.pdf> – viimati vaadatud 30.05.2010.
8. Manuel Luqueand, Herbert Voß. 3D views with pst-vue3d, 2007. <http://www.ctan.org/tex-archive/graphics/pstricks/contrib/pst-vue3d/pst-vue3d-doc.pdf> – viimati vaadatud 30.05.2010.
9. Herbert Voß. Pst-3dplot documentation, 2010. <http://ftp.linux.ee/pub/tex-archive/graphics/pstricks/contrib/pst-3dplot/pst-3dplot-doc.pdf> – viimati vaadatud 30.05.2010.
10. Denis Girou, Herbert Voß. The pst-ob3d package, 2007. <http://ftp.linux.ee/pub/tex-archive/graphics/pstricks/contrib/pst-ob3d/pst-ob3d.pdf> – viimati vaadatud 30.05.2010.
11. Jean-Paul Vignault, Manuel Luque, Arnaud Schmittbuhl. Pst-solides3d: The Documentation – The Basics, 2010. <http://ftp.linux.ee/pub/tex-archive/graphics/pstricks/contrib/pst-solides3d/pst-solides3d-doc.pdf> – viimati vaadatud 30.05.2010.
12. Manuel Luque. Prototype-vehicule, 2008. http://melusine.eu.org/lab/bpst/pst-solides3d/objfusion/prototype_vehicule.pst – viimati vaadatud 30.05.2010.
13. Till Tantau. The TikZ and PGF Packages, 2008. <http://ftp.linux.ee/pub/tex-archive/graphics/pgf/base/doc/generic/pgf/pgfmanual.pdf> – viimati vaadatud 31.05.2010.
14. Jeff Hein. The tikz-3dplot Package, 2010. http://ftp.linux.ee/pub/tex-archive/graphics/pgf/contrib/tikz-3dplot/tikz-3dplot_documentation.pdf – viimati vaadatud 31.05.2010.
15. Bryan Henderson. Netpbm home page, 2008. <http://netpbm.sourceforge.net/> - viimati vaadatud 31.05.2010.
16. René K. Müller. GIFMerge, 2001. <http://www.the-labs.com/GIFMerge/> - viimati vaadatud 31.05.2010.
17. SWFTTools. <http://www.swftools.org/> - viimati vaadatud 31.05.2010.

18. ImageMagick. <http://www.imagemagick.org/script/index.php> – viimati vaadatud 31.05.2010.
19. Examples of slides class.
<http://amath.colorado.edu/documentation/LaTeX/reference/slides/> - viimati vaadatud 01.06.2010.
20. Frederic Goualard. Manual for the prosper class.
<http://amath.colorado.edu/documentation/LaTeX/prosper/0/prosper-doc.pdf> – viimati vaadatud 01.06.2010.
21. Till Tantau. The Beamer class, 2007. <http://ftp.linux.ee/pub/tex-archive/macros/latex/contrib/beamer/doc/beameruserguide.pdf> – viimati vaadatud 01.06.2010.
22. Hans Fr. Nordhaug. TeXPower home page, 2009. <http://texpower.sourceforge.net/> - viimati vaadatud 01.06.2010.
23. Stephan Lehmké, Hans Fr Nordhaug. The TeXPower bundle, 2005.
<http://ftp.linux.ee/pub/tex-archive/macros/latex/contrib/texpower/manual.pdf> – viimati vaadatud 01.06.2010.
24. PPower4 home page. <http://www-sp.iti.informatik.tu-darmstadt.de/software/ppower4/> - viimati vaadatud 01.06.2010.
25. Klaus Guntermann, Christian Spannagel. PPower4 Manual, 2002.
<http://ftp.linux.ee/pub/tex-archive/support/ppower4/manual.pdf> – viimati vaadatud 01.06.2010.
26. Alexander Grahn. The Animate package, 2010. <http://ftp.linux.ee/pub/tex-archive/macros/latex/contrib/animate/animate.pdf> – viimati vaadatud 01.06.2010.
27. Alexander Grahn. The Movie15 package, 2009. <http://ftp.linux.ee/pub/tex-archive/macros/latex/contrib/movie15/doc/movie15.pdf> – viimati vaadatud 01.06.2010.
28. Timo Hartmann. Flashmovie package, 2010. <http://www.ctan.org/tex-archive/macros/latex/contrib/flashmovie/> - viimati vaadatud 01.06.2010.
29. Adobe Acrobat family: Product comparison, 2009.
<http://www.adobe.com/products/acrobat/matrix.html> – viimati vaadatud 01.06.2010.
30. OpenOffice Impress User Guide.
http://www.linuxtopia.org/online_books/office_guides/openoffice_impress_guide/ - viimati vaadatud 01.06.2010.
31. Microsoft Office PowerPoint 2010 Beta.
<http://www.microsoft.com/office/2010/en/powerpoint/default.aspx> – viimati vaadatud 01.06.2010.

Lisad

Lisades mainitud failid leiab http://comserv.cs.ut.ee/forms/ati_report/index.php?year=2010 leheküljelt kokkupakituna ZIP formaadis. Kaasas on nii TeX algfailid, kui ka nendega koostatud PDF formaadis lõpptulemus.

Lisa 1

Tõeväärtustabeli animatsiooni koostamine ning *animate* paketi abil selle paigutamine ühele PDF lehele. Animatsiooni käigus lisandub tabelisse tõeväärtusi. Kaasas on järgnevad failid:

[Lisa1.tex](#) – fail koodiga animatsiooni koostamiseks.

[Lisa2.pdf](#) – tulemusfail koostatud animatsiooniga.

Lisa 2

Silindri animatsiooni kaadrifailide koostamine PSTricks ning Pst-solides3d pakettide abiga. *Animate* laiendust on kasutatud terviku kokkupanekuks kaadrifailidest. Animatsiooni tulemuseks on silindri liikumine nii, et seda näeb erinevate nurkade alt.

Kaasasolevad failid:

[Lisa2-kaadrid.tex](#) – fail koodiga kaadrite koostamiseks.

[Lisa2-kaadrid.pdf](#) – tulemusfail, kus iga kaader on omal lehel.

[Lisa2-animatsioon.tex](#) – fail koodiga kaadritest terviku kokkupanemiseks.

[Lisa2-animatsioon.pdf](#) – tulemusfail, kus on terviklik animatsioon.