

TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Hannes Saariste
Metsatulekahjude simulatsioon reaalses
Bakalaureusetöö (9 EAP)

Juhendaja: Jaanus Jaggo, MSc

Metsatulekahjude simulatsioon reaajas

Lühikokkuvõte:

Käesolevas bakalaureusetöös antakse ülevaade metsatulekahjude tekke, tule leviku ja tüüpide kohta. Nende andmete põhjal implementeeritakse mängumootoris Unity metsatulekahju simulatsioon, mis näitab reaajas tule levikut, arvestades erinevaid tegureid nagu näiteks materjali niiskus, temperatuur, maastiku tõus ja langus ning tuule kiirus. Simulatsiooni sisendiks on kasutajal võimalik sisestada laserskanneerimise ja satelliitfoto andmed, mille põhjal genereeritakse vastav maastik. Samuti võimaldab programm erinevaid maastikke genereerida protseduuriliselt. Rakendust edasi arendades saab sellest teha näiteks õpetliku mängu, mis näitaks, millist ohtu metsatulekahju kujutab. Samuti sobiks see ka reaalse olukordade läbimängimiseks ning tuletõrjajate väljaõppeks ja aitaks koostada paremaid plaane metsatulekahjude avastamiseks ja kustutamiseks.

Võtmesõnad:

Metsatulekahjud, protseduuriline genereerimine, arvutigraafika

CERCS: P175: Informaatika, süsteemiteooria

Real-time wildfire simulation

Abstract:

This bachelor thesis gives an overview of different wildfire types, causes of ignition and spreading. All this data will be used for making a simulation in Unity game engine that can show the spreading of wildfire in real-time depending on humidity, temperature, landscape and wind speed. User can input laser measuring data images and satellite photo to generate real landscape. The simulation can also procedurally generate random landscape. In case of further development, it can be used as instructive game to illustrate how big of a threat wildfire can be. With sufficient improvements it can be used for making better plans for real-life situations to detect and extinguish forest fires.

Keywords:

Wildfires, procedural generation, computer graphics

CERCS: P175: Informatics, systems theory

Sisukord

1.	Sissejuhatus.....	5
2.	Metsatulekahjud üldiselt	6
2.1	Metsatulekahjude levik.....	6
2.2	Tule leviku mudelid.....	8
3.	Tulekahjude simulatsiooni tööriistad	10
3.1	Wildfire Analyst.....	11
3.2	FARSITE	12
4.	Implementatsioon.....	13
4.1	Reaalsete andmetega loodud maastik.....	13
4.1.1	Laserskanneerimis andmete kasutamine	14
4.2	Protseduuriline maastiku genereerimine.....	17
4.3	Tule leviku implementatsioon.....	18
4.3.1	Varjutajas põlengu arvutamine	18
4.3.2	Andmetekstuuride kasutamine	19
4.3.3	Põlengu sãmplimine rakuautomaadi abil	20
4.3.4	Ringikujuline sãmplimine	21
4.4	Vãlistegurite arvestamine	23
4.4.1	Perlini mura	24
4.5	Kasutajaliides ja visuaalne pool.....	24
4.5.1	Jãreltõõtlus.....	26
5.	Simulatsiooni jaoks arendatud valemid	27
5.1	Temperatuur	27
5.2	Niiskus	28
5.3	Topograafia	29
5.4	Tuul.....	30
6.	Tãpsus, testimine ja edasiarendused.....	31
6.1	Jõudlus	32
7.	Kokkuvõte.....	33
8.	Viidatud kirjandus	34
Lisad	36
I.	Mõisted ja terminid	36
II.	Rakenduse pildid.....	37
III.	Rakenduse kasutamine	38

IV. Lähtekood	39
V. Litsents	40

1. Sissejuhatus

Metsatulekahjud on maailmas üha suuremaks probleemiks. Globaalse soojenemise tõttu langeb niiskuse tase ja sademete kogus, mille tulemusena muutuvad kuivad alad veel kuivemaks. Kõrgemate temperatuuride tõttu pikeneb ka metsatulekahjude hooaeg, tulekahjud on suuremad ja kestavad kauem [3].

Käesoleva bakalaureusetöö eesmärgiks on luua simulatsioon, mis näitab reaalses metsatulekahju levikut, alustades väikesest lõkkest, mis kasvab hektarite suuruseks põlenguks. Selle rakenduse eesmärk on tõsta inimeste teadlikkust metsatulekahjude ohtlikkusest ja kaasata seeläbi inimesi tulekahjude ennetusele. Kuna rakendus on tehtud mängumootoris Unity, saab seda edasi arendades luua õpetliku mängu lastele, või kasutada elulise olukorra katsetuste ja tulekahjude piiramis- ja kustutusplaanide loomiseks.

Simulatsiooni käivitamisel genereeritakse kasutaja poolt ette antud laserskanneerimise ja satelliitfotode põhjal kõrguskaart. Nende puudumisel saab luua kõrguskaarti ka protseduuriliselt. Selle kõrguskaardi põhjal koostatakse maastiku geomeetria, millele kuvatakse kas kasutaja poolt määratud satelliitfoto või ettemääratud maastiku tekstuur. Tule levikut hakatakse järkjärguliselt arvutama graafikakaardi varjutajas. Graafikakaardi paralleelne arhitektuur võimaldab seda simulatsiooni arvutada kordades kiiremini, kui protsessoril. Varjutaja arvutab igal sammul, kui palju on metsatulekahju möödunud aja jooksul edasi liikunud ning renderdab uue seisu tekstuurile. See vahepealne seis tehakse ka kasutaja jaoks nähtavaks, kuvades see kolmemõõtmelisele maastikule.

Peatükis 2 kirjeldatakse, kuidas metsatulekahjud levivad looduses. Peatükis 3 antakse lühike ülevaade varasematest tulekahju simuleerimistöriistadest. Peatükis 4 kirjeldatakse implementeeritud programmi, selle tööd, loogikat ja erinevaid lähenemisviise. Peatükis 5 antakse täpsem ülevaade simulatsiooni jaoks arendatud valemitest. Peatükis 6 on välja toodud mudelite ja simulatsiooni võimalikud veaprotsendid, jõudluse testimise tulemused erinevate arvutite peal ning kirjeldatud edasiarenduse võimalusi.

2. Metsatulekahjud üldiselt

Metsatulekahju on kontrollimatu tulekahju põleva taimestikuga maapiirkonnas. Metsatulekahjusid on erinevat liiki, näiteks pinnatuli, mis levib mööda maapinna alusmaterjali, ladvatuli, mis kandub ühelt puult teisele mööda puuvõrasid ja lendtuli, mille puhul õhus lendav põlev materjal tekitab uusi põlengualgeid. Metsatulekahju erineb teistest tulekahjudest oma väga suure põlenguala poolest. Põleng levib tohutu kiirusega, ning võib tuulte tõttu ootamatult leviku suunda muuta [1].

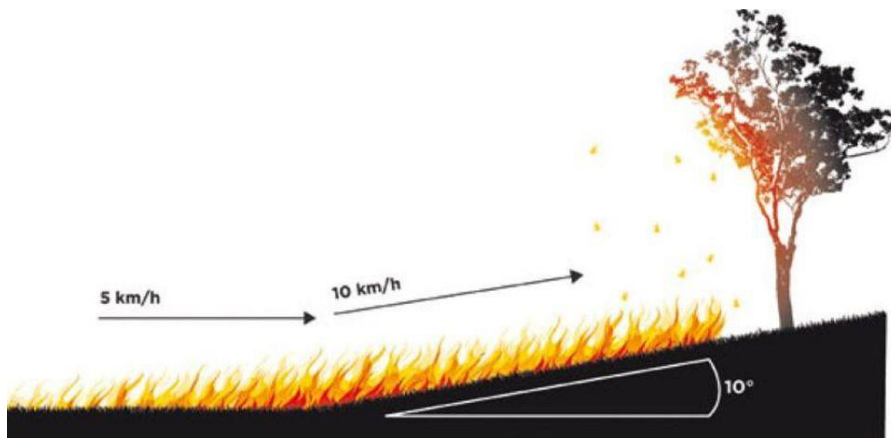
Selleks, et põleng alguse saaks, piisab pikselöögist või päikesest tulenevast kuumusest. Kevin Bonsor väidab oma artiklis [4], et isegi rongi rataste ja raudteerööbaste vahel tekkinud sädemest piisab, et põleng alguse saaks. Suurem osa metsatulekahjusid algab siiski inimeste enda hooletusest, olgu selleks kas maha visatud suitsukoni, hooletult tehtud lõke või ilutulestiku kasutus eirates ohutusnõudeid [4].

2.1 Metsatulekahjude levik

Puitmaterjali süttimistemperatuur on 300 kraadi. Sellisel temperatuuril vabanev süsivesinik seguneb õhuga ja süttib põlema. Põlemiseks on vajalik peamiselt kolm komponenti (kujutatud joonisel 5): Põlemiseks vajalik kütus (puit, lehed, hein), hapnik ja kuumus. Kui neist kolmest ühe elimineerida on tulekahju tekkimine või selle olemasolul jätkumine võimatu [4].

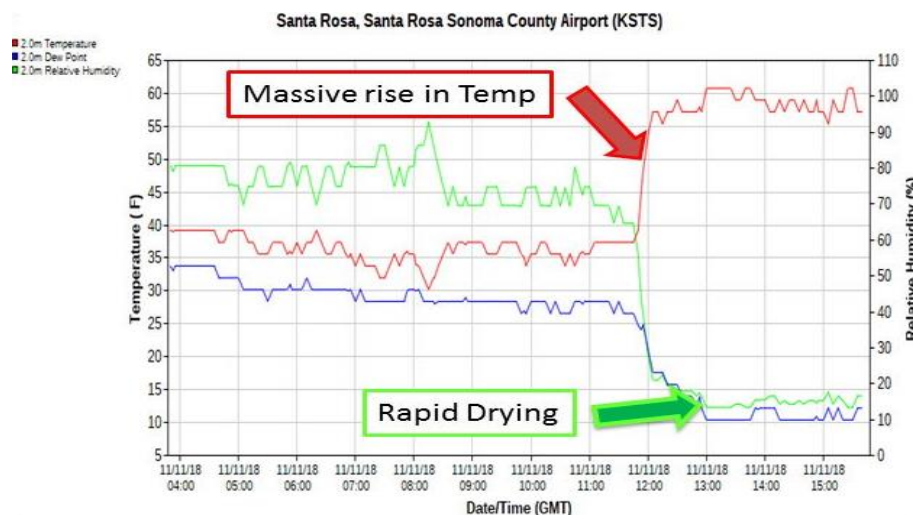
Kui põlemine toimub on jällegi kolm põhilist tegurit, mis moodustavad põlengu keskkonna: Kütus, ilmastik ja topograafia, mis on välja toodud joonisel 2. Kütuseks on nii puit, põõsad, hein, kui ka elumajad. Mida kuivem on materjal, seda kiiremini see ka süttib. Niiskus sõltub peamiselt ilmastikust, olles vihmaperioodil kõrgem ning kuival ajal madalam.

Ülesmäge levib tuli kiiremini kui allamäge. Suure tõusu puhul mõjutab kuumus suuremat ala ja seega süttib tule ette jääv materjal kiiremini [4]. Piisab vaid 10 kraadisest tõusust, et tule levik kaks korda kiireneks. Seda on näha ka järgneval joonisel 1.



Joonis 1. Tule levik tõusul [6].

Tuul on põlengu levikule kõige suurema mõjuga. See mõjutab leegi suunda ja nii süttib alla tuult olev materjal kuumuse tõttu kiiremini. Lisaks võib tuul kohale tuua soojad õhuvoolud, mis veel omakorda taimsetikku kuivatab. Seetõttu on põlengu levik veel kiirem [18]. Muutuste kiirust näitab allpool toodud NWS Bay Area twitteri postitus graafikust joonisel 2, kus tuule suuna muutus tõi kaasa 15 kraadise temperatuuri tõusu ja õhuniiskus langes üle 60% [19]. Tulekahjust tulevast kuumusest tekivad tuulekeerised, mis suurendavad põlengu intensiivsust rohke hapniku pealevooluga. Selliste õhuvooludega levivad õhu kaudu põlevad lehed ja oksad, mis võivad süüdata uue tule kuni 100m kaugusel ületades lagedaid alasid ja jõgesid [4].

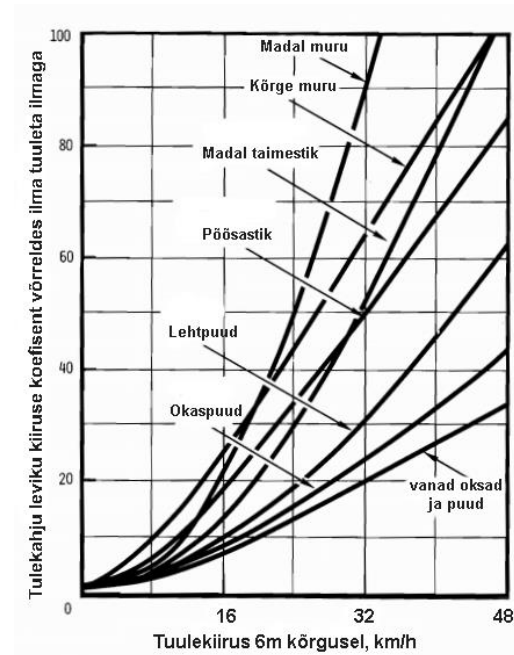


Joonis 2. Järsk temperatuuri kasv ja õhuniiskuse langus [20].

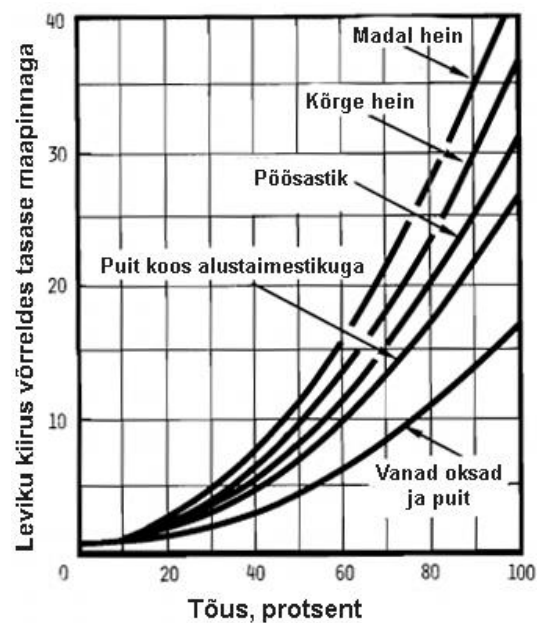
2.2 Tule leviku mudelid

Richard C. Rothermel, teatud ka kui Dick Rothermel on insener, kes arendas ühe seni kõige kasutatava tule leviku mudeli. On tehtud mitmeid alternatiivseid mudeleid, kuid just Rothermeli mudel on tänapäevani laialdaselt kasutusel metsatulekahjude kustutamise planeerimisel [21].

Järgnevalt on välja toodud kaks graafikut, mis näitavad tuule ning maastiku tõusu ja languse mõju tule levikule. Vastavad graafikud on loodud Rothermel'i mudeli põhjal Frank A. Albini poolt [2]. Nende graafikute andmeid on kasutatud lõputöö raames valminud simulatsiooni loogika koostamisel. Joonisel 3 kujutatud graafik näitab tule liikumise kiirenemist tõusu kasvades, erinevate materjalide kohta. Graafik joonisel 4 näitab kui palju kiireneb tule levik sõltuvalt tuule kiirusest.

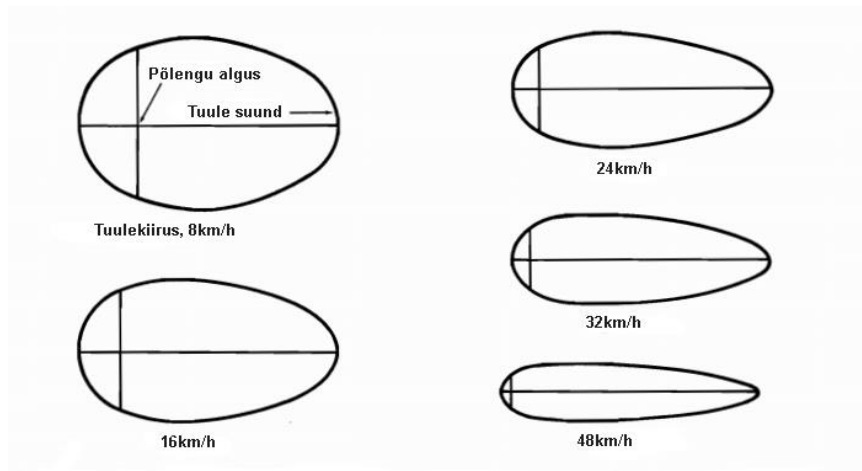


Joonis 3. Tule leviku kiirus sõltuvust tuule kiirusest [2].



Joonis 4. Tule leviku kiirus tulenevalt maastiku tõusust [2].

Joonisel 5 on välja toodud metsatulekahju leviku maa-ala kuju ja selle muutus olenevalt tuule kiirusest. Tuli saab alguse kahe sirge ristumiskohast ja tuule suund on vasakult paremale. Sellist piklikku muna kujulist ala on täheldatud isegi väga suurte tulekahjude korral [2]. Taoline põlengu ala kuju tekib tulenevalt tuule mõjust, kus vastu tuult põlengu levik aeglustub ja alla tuult oluliselt kiireneb.



Joonis 5. Tule leviku maa-ala kuju sõltuvalt tuulest [2].

3. Tulekahjude simulatsiooni tööriistad

Metsapõlengutega toimetulekuks on tähtis teada kuhu ja kui kiiresti põleng levib. Selle info põhjal on parem planeerida, mis piirkondi ja mis järjekorras tuleks evakueerida. Saab umbes arvestada, mis kaugusele tuleks teha kütuse vabasid tsoone põlengu peatamiseks nii, et need enne põlengut valmis jõutaks. Kindlasti peab arvestama, et ükski mudel ja andmekogum pole piisavalt täpne, mis suudaks arvesse võtta kõiki võimalikke tegureid, nende koosmõju ja looduse eripärasid. Seega peab alati arvestama võimaliku veaga.

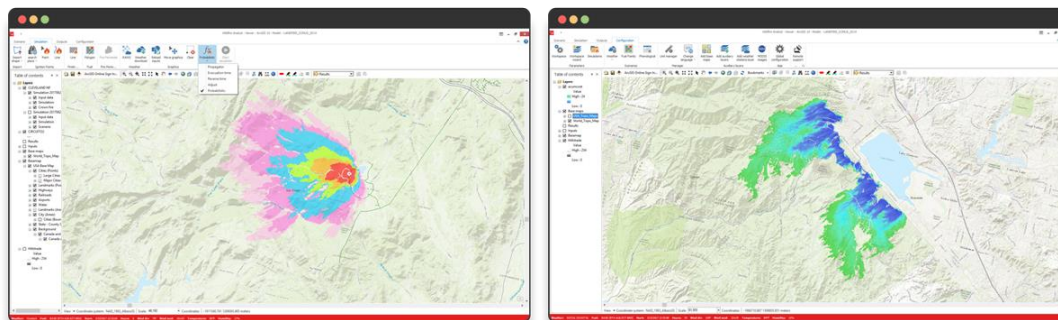
Siin on nimekiri mõningatest olemasolevatest tulekahju simulaatoritest [22]:

- Prometheus – Kanada riiklik avatud lähtekoodiga tarkvara.
- Phoenix – Austraalias arendatud tulekahju riskijuhtimis tööriist.
- FireStation – Raster baasil Geinfosüsteemi platvorm.
- Hfire – Raster baasil ruumiline tulekahju leviku tööriist.
- Wildfire Analyst – Reaalajas tulekahju modelleerimis süsteem.
- FARSITE – Tule leviku modelleerimis süsteem.

Selles peatükis antakse lühike ülevaade levinud metsatulekahjude simulatsiooni tööriistadest, milleks on Wildfire Analyst ja FARSITE.

3.1 Wildfire Analyst

Tegu on uuemat tüüpi programmiga, mis sisaldab endas paljusid erinevaid tööriistu. Lisaks tavapärasele põlengu simuleerimisele on võimalik planeerida materjali tahtlikku põletamist tule peatamiseks, teha riskianalüüsi ja tulekahju järgset uurimist. Toetatud on ka integreerimine kolmanda osapoole tarkvaraga. Selle programmi abil saab arvutada põlengu levikut, on võimalik integreerida ilmateenuseid pidevate uuenduste jaoks ja toetatud on ka kütuste kaardistamine. Wildfire Analyst ei vaja kasutajalt põhjalikke metsatulekahjude teadmisi, erinevalt näiteks FARSITE-st. Andmeid pole vaja eelnevalt käsitsi töödelda, mis võimaldab kasutajal kiiremini simulatsiooni tulemusi näha. Järgnevalt on joonisel 6 mõned näited, kuidas tarkvara visuaalselt välja näeb [15].



Joonis 6. Wildfire Analyst kuvatõmmised

3.2 FARSITE

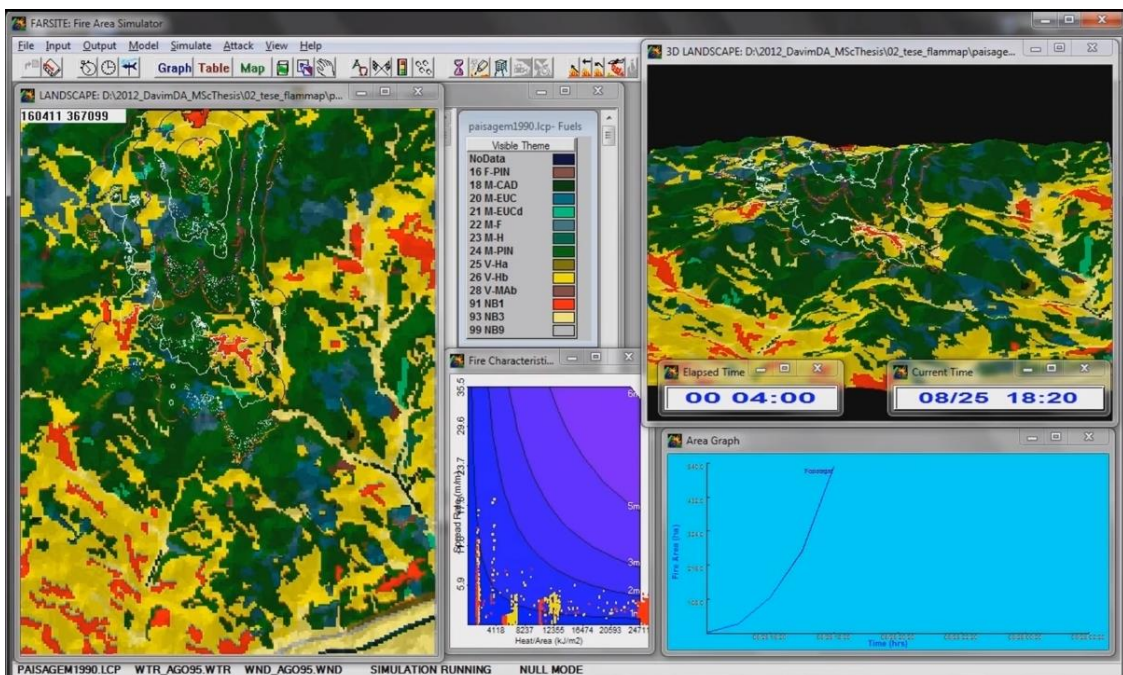
FARSITE on tulekahju leviku modelleerimistarkvara, mida kasutavad U.S. Forest Service ja National Park Service, et simuleerida metsatulekahjude levikut maastikul. Suunatud on see kasutajatele, kes on teadlikud metsatulekahjude, kütuste, ilmastiku ja topograafiaga seotud terminoloogiast. Oma keerukuse tõttu eeldab FARSITE korralikku väljaõpet. Programmi sisendiks on eri tüüpi andmefailid, mis pole laialt levinud ja on väga valdkonna-spetsiifilised [16].

Kasutusel on järgmised tulekahju mudelid [16]:

- Rothermel'i (1972) Pinna tulekahju mudel.
- Van Wagner'i (1977) ladvatule algatus mudel.
- Rothermel'i (1991) ladvatule leviku mudel.
- Albini (1979) Põleva materjali lendamis mudel.
- Nelson'i (2000) surnud taimeistiku niiskuse mudel

FARSITE on deterministiline modelleerimissüsteem, seega on tulemused otseses seoses sisenditega. Tarkvara saab kasutada nii põlengu kustutamise ja evakuatsiooni planeerimiseks kui ka lihtsalt “mis siis kui” olukordade katsetamiseks [16].

Tegu on küll realses olukorras pädeva tarkvaraga tulekahjudega toimetulekuks, kuid on aegunud. Seda on näha nii kasutajaliidese aegunud visuaalide, kui ka dokumentatsiooni Windows 2000 soovitusel. Järgnevalt on joonisel 7 näha kuvatõmmist FARSITE rakendusest. Vasakul üleval on heleda joonega näha tule leviku ala, põlengu algusest on möödunud 4 tundi.



Joonis 7. FARSITE kuvatõmmis

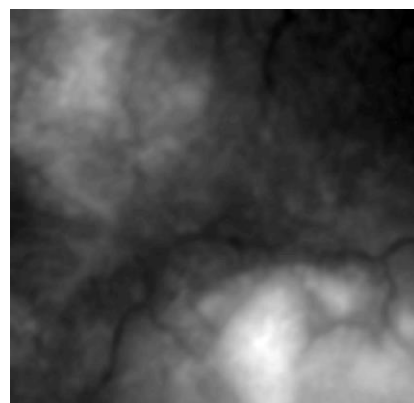
4. Implementatsioon

Selles peatükis antakse ülevaade töö raames valminud simulatsiooni osadest:

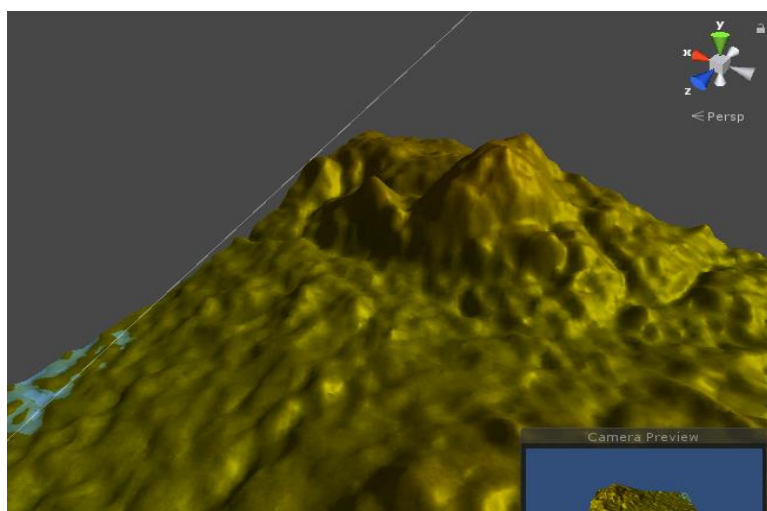
- Realistlike andmetega maastik.
 - Laserskanneerimise andmete kasutamine ja töötlemine
- Protseduuriline maastiku genereerimine.
- Tule leviku implementatsioon.
 - Rakuautomaadi (*ingl. k Cellular Automaton*) stiilis lähenemine.
 - Andmetekstuuride kasutamine.
 - Ringikujuline sãmplimine.
- Välistegurite mõju arvestamine.
- Kasutajaliides.

4.1 Reaalsete andmetega loodud maastik

Lisaks protseduurilise maastiku genereerimisele on võimalik kasutada ka suvalist kõrguskaarti. Nii võib katsetada tulekahju levikut ükskõik mis paigas, kasvõi enda kodu ümbruses. Vastavad andmeid saab Terrain-Party¹ lehelt. Need kõrguskaardid on aga otse kasutamise jaoks liiga mürarikkad. Seetõttu on ka tulemuseks saadud maastik liiga künklik. Selle parandamiseks on vajalik mõõdukas udustamine näiteks mõne pilditöötlus programmiga. Pilt vajab ka värvide skaalal laiali jaotamist, et andmed oleks simulatsioonile sobival kujul. Joonisel 10 on näha Eesti kaardilt võetud ala, kus paremal all on Suur Munamägi. Saadud maastiku on näha joonisel 11.



Joonis 10. Kõrguskaart Eesti kaardi alast, paremal all Suur Munamägi.

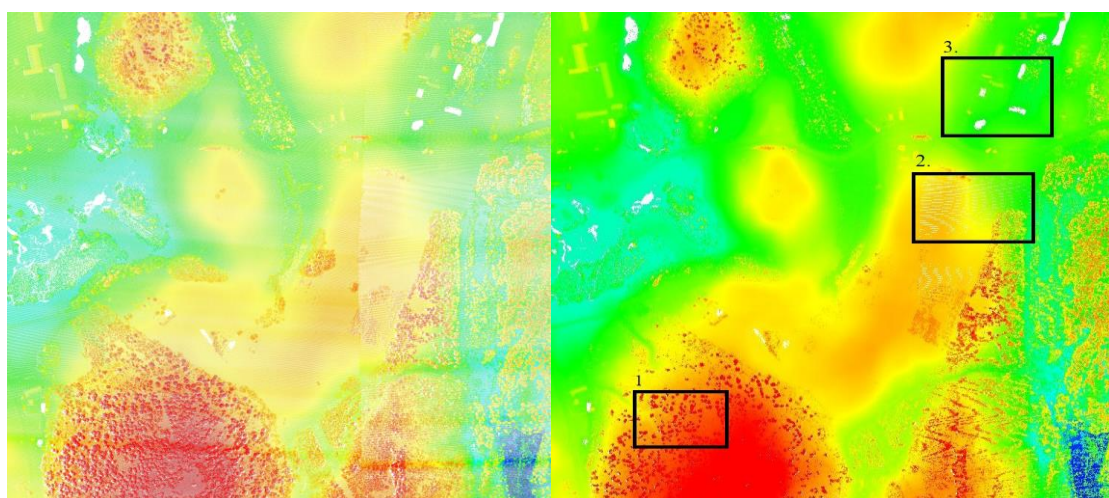


Joonis 11. Simulatsiooni maastik Suurest Munamäest.

¹ <http://terrain.party/>

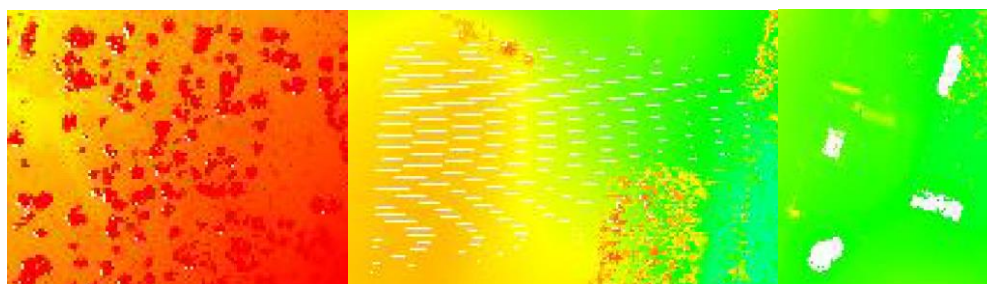
4.1.1 Laserskanneerimis andmete kasutamine

Simulatsioonis on toetatud ka laserskanneeritud kaartide kasutamine. Igal aastal laserskanneeritakse Maa-ameti poolt uuesti umbes neljandik tervest Eestist. Need kaardid on tasuta kättesaadavad Maa-ameti lehel [10]. Nende andmete põhjal saab leida kõrguskaardi mis tahes Eesti piirkonna kohta. Joonisel 12 on näha kahte aeroskanneerimise tulemust Haanja kandist, kus all vasakul nurgas nähtav punane osa kujutab Suurt Munamäge.



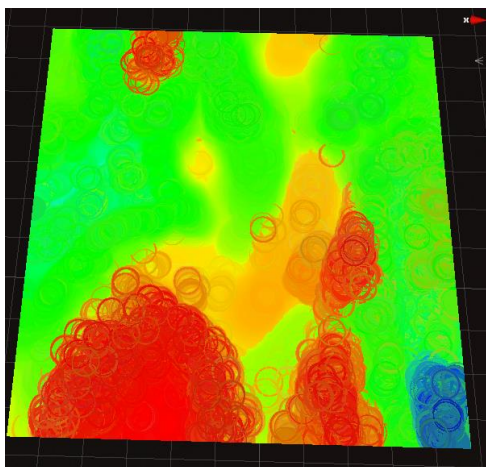
Joonis 12. vasakul tavalennu ja paremal madallennu aeroskanneerimise tulemus Haanja kandist [10]. Kastid 1, 2 ja 3 on suurendatult joonisel 13.

Lähemal uurimisel on näha, et andmetes on palju vigu. Joonisel 13 on mõningad näited, kuidas maastiku skanneerimisel ei saa sensor alati tagasisidet ning kuna antud punkti koha kõrgusandmed puuduvad, siis on see tähistatud valge värviga.

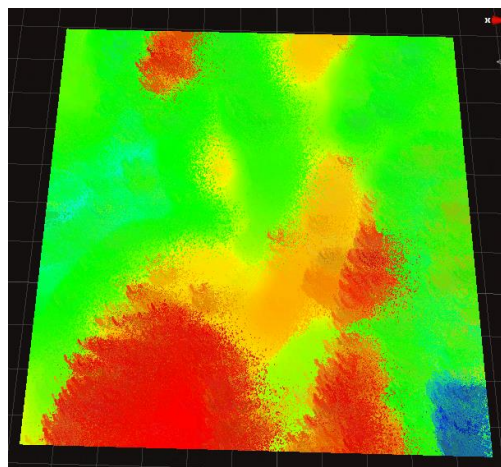


Joonis 13. Aeroskanneerimise andmete vead, suurendatud jooniselt 10.

Vigade parandamiseks on käesolevas töös loodud varjutaja, mis täidab ära puudu olevad pikslid, ümbritsevate värvidega. Samuti ühtlustab varjutaja Iga piksli lõpliku värvi määramiseks vaadatakse selle lähedal asetsevaid piksleid ning leitakse kõrguste keskmine, mis ühtlustab mõõteandmed ja annab simulatsiooni jaoks sujuva kõrguskaardi. Maapinna kõrguse skaalal on värvitoonid alati kahe värvi kombinatsioon. Seega kontrollitakse iga punkti puhul ühe värvi puudumist. Kui esinevad kõik kolm värvi, siis seda punkti samplimisel ei arvestata. Seejärel rakendati samplimis raadiusele müra, et võtta punkte ringi seest erineva raadiusega. Endiselt jäi tulemus, mida on näha joonisel 15, alla rahuldava.

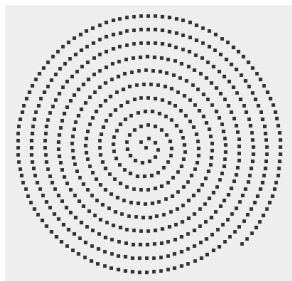


Joonis 14. Ringis samplimine.

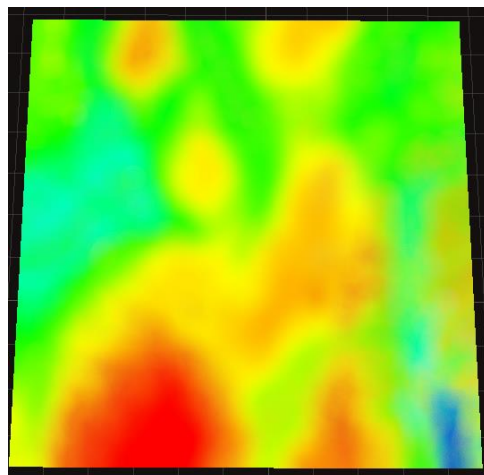


Joonis 15. Ringis samplimine müraga.

Tunduvalt parema tulemuse andis piksli ümbruse samplimine spiraalselt (näha joonisel 16). Punktide leidmiseks liigutakse keskpunktist iga punktiga mingi arvu võrra välja ja mingi kraadi võrra edasi. Antud viisil ei teki ringi kujuga vigu ja nagu näha joonisel 17, tuleb kõrguskaart ühtlasem. Vastav loogika on päris Pete Kirkham'i Stack Overflow postitusest [11].

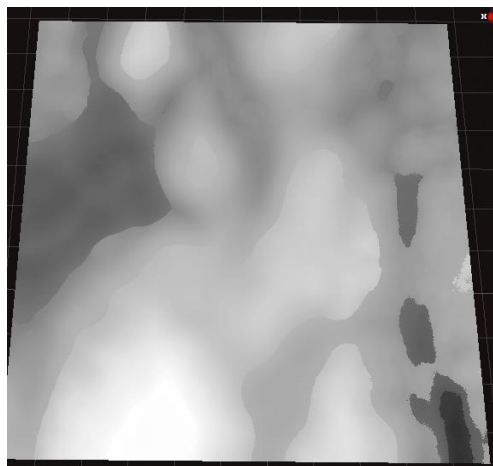


Joonis 16. Spiraalne punktide arvutamine [11].



Joonis 17. Spiraalse samplimise tulemus.

Lõpetuseks teisendatakse värvidega tähistatud kõrgusinfo must-valgeks kõrguskaardiks. Selleks saab varjutaja ette 1x256 piksli suuruse tekstuuri, mis on kujutatud joonisel 19, millega võrreldakse kõrguskaardi piksli värvi. Võrdlemiseks leitakse tekstuuride värvikanalite vahe absoluutväärtuste summa. Kõige väiksema erinevusega piksel määrab ära mis tooni saab piksel olema skaalal nullist üheni, kus null vastab mustale ja üks valgele. Joonisel 18 on näha lõpptulemusena valminud must-valget kõrguskaarti, mille põhjal genereeritakse simulatsiooni maastik.



Joonis 18. Must-valge kõrguskaart.

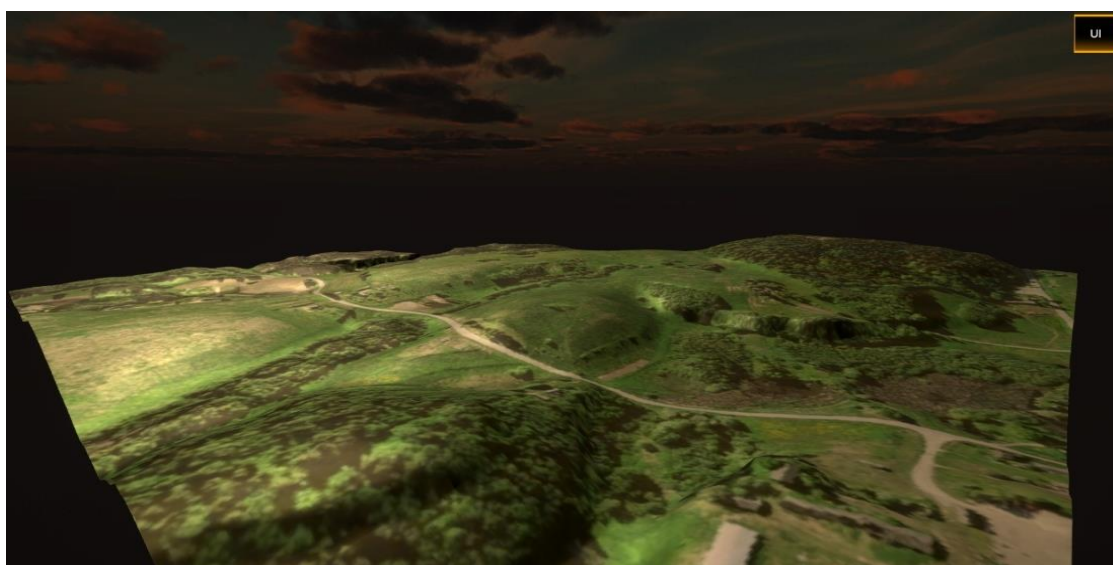


Joonis 19. kõrguse skaala.



Joonis 20. Sateliitfoto näidis [10].

Viimasena kuvatakse kõrguskaardi põhjal genereeritud maastiku geomeetriaal sateliitfoto (näide joonisel 20) või selle olemasolu puudumisel standard maastiku tekstuur. Näide lõplikust maastikust Haanja piirkonnast on näha järgneval joonisel 21.

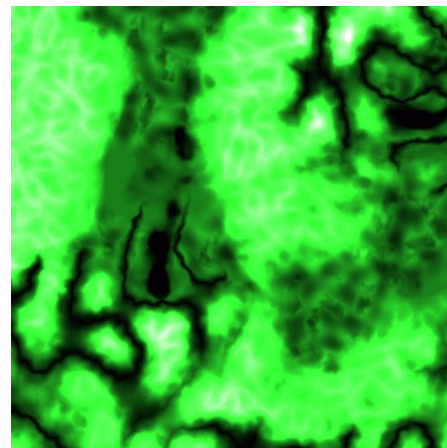


Joonis 21. Simulatsiooni maastik Haanja piirkonnast.

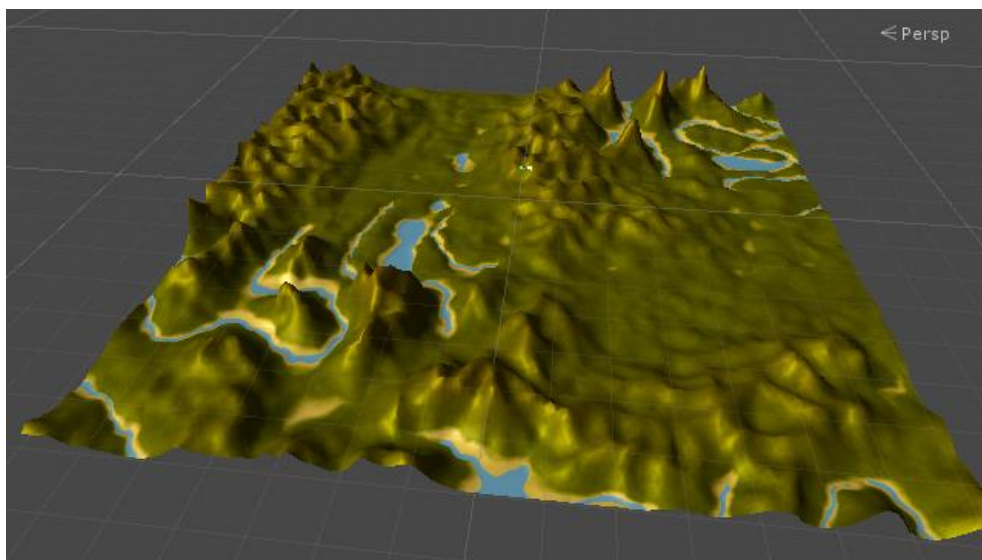
4.2 Protseduuriline maastiku genereerimine

Simulatsioon võimaldab protseduuriliselt maastiku genereerimist. Selleks on kasutatud Jaanus Jaggo loodud maastiku-generaatorit.

Esmalt genereeritakse varjutajas maastiku kõrguskaart (näidatud joonisel 8) ja salvestatakse see tekstuurile. See tekstuur kujutab endast kõrguskaarti, mille värvid määravad ära vastavas punktis oleva maastiku kõrguse. Järgnevalt koostatakse maastiku geomeetria, sãmplides selleks kõrguskaarti iga geomeetria serva kohalt. Kõrguskaardi tekstuur antakse edasi ka loodud geomeetria tekstuuriks. Joonisel 8 nähtava kõrguskaarti põhjal genereeritud maastik on näha joonisel 9.



Joonis 8. Protseduuriliselt genereeritud kõrguskaart



Joonis 9. Protseduuriliselt genereeritud maastik

4.3 Tule leviku implementatsioon

Selles alampeatükis kirjeldatakse varjutaja kasutamise eelistest võrreldes protsessoris andmete töötlemisega, räägitakse andmetekstuuride kasutamisest ja erineval viisil punktide sãmplimisest. Tule leviku arvutamiseks on võrreldud erinevaid võimalusi.

4.3.1 Varjutajas põlengu arvutamine

Arvuti puhul on võimalik arvutusi teha kas protsessori või videokaardiga. Protsessor suudab tegeleda paljude eri ülesannetega, seejuures videokaart on pigem suunatud mõnele kindlale ülesandele. See tuleneb nende arhitektuurist, kus protsessori tuumade arv jääb tavakasutaja jaoks alla 24, kuid videokaart kasutab tuhandeid väiksemaid tuumasid paralleelarvutuste tegemiseks. Arvutigraafika alaste arvutuse jaoks, kus tuleb näiteks iga piksli puhul teha palju väikseid kalkulatsioone, suudab videokaart teha oma tööd 50-100 korda kiiremini. Järjest rohkem rakendatakse videokaarte ka näiteks masinõppes [18].

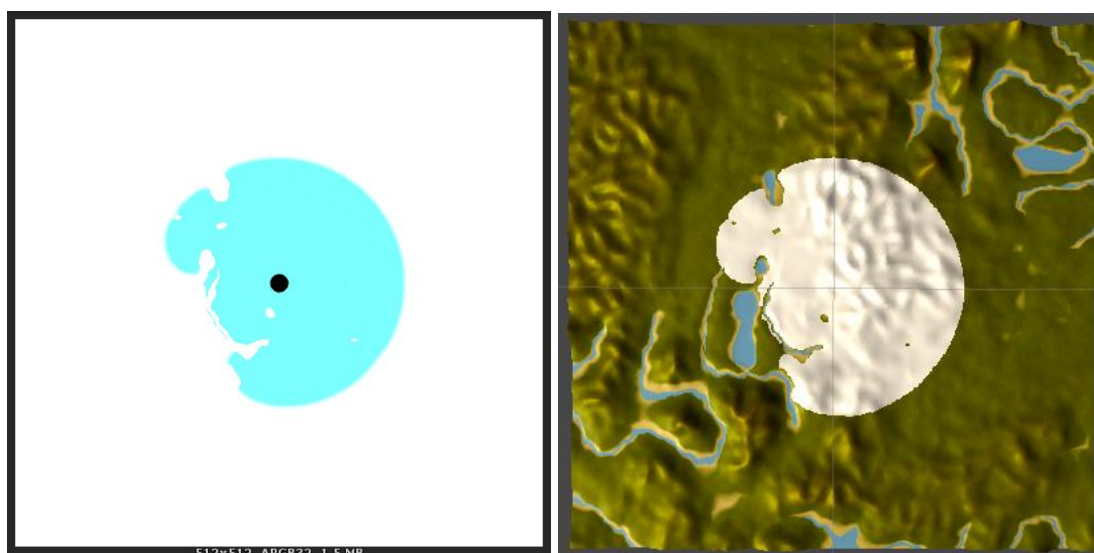
Algselt olid varjutajad mõeldud valgustuse arvutamiseks, kuid hilisemalt kasutusel ka erinevate efektide, järeltõtluse või lihtsalt tavaarvutuste tegemiseks. Varjutajaid kasutatakse suure hulga andmete kiireks muutmiseks, näiteks ekraani iga piksli arvutamiseks. Tänu paralleelarvutustele suurendatakse arvutuste läbilaskevõimet [17].

Eelnevalt nimetatud põhjustel on simulatsioonis arvutuste tegemiseks eelistatud varjutajate kasutamist. Nii võib teha igas kaadris iga piksli puhul tohutult arvutusi ja seejuures ei teki probleeme jõudlusega. Sujuva animatsiooni loomiseks tuleb arvutigraafika programmides renderdada vähemalt 60 kaadrit sekundis, sest see on ka tüüpilise kuvari värskendussagedus. Iga kaadri renderdamisel arvutatakse kõik andmetekstuurid uuesti (Hetkel on kasutusel 512x512 piksli suurused andmetekstuurid, kus omakorda iga piksli väärtuse arvutamiseks sãmplitakse mitusadat ümbritsevat pikslit). Näiteks põlengu andmetekstuuri jaoks tehakse iga kaader 512x512x100 ehk üle 26 miljoni arvutuse. Protsessor jääb selliste arvutuste jaoks liiga aeglaseks ja kaadrisagedus oleks madal. Samas videokaart on just selle jaoks loodud ja jõudlusega probleeme ei teki.

4.3.2 Andmetekstuuride kasutamine

Varjutajale saab andmeid ette anda muutujate ja tekstuuridena. Tule leviku andmete hoidmiseks salvestatakse need tekstuuri erinevatesse värvikanalitesse. Samuti hoitakse tekstuurides keskkonna andmeid, nagu temperatuur ja niiskus. Neid olemasolevaid tekstuure sãmplides otsustab varjutaja, valemide kasutades, mida iga piksliga kaadri renderdamisel teha.

Põlengu liikumist mõjutavate tegurite info salvestamiseks on kasutusel erinevad kontroll tekstuurid, mis sisaldavad infot värvikanalites. Selleks on kasutusel neli kanalit: punane, roheline, sinine ning läbipaistvus. Iga kaadri renderdamisel arvestatakse ette antud tekstuure ja varjutaja valemide ja muutujaid. Vastava info põhjal tehakse otsus, kui palju on vaja põlengu andmetekstuuris piksli väärtust vähendada ehk põlengu mõttes, kui palju on tuli ühe kaadri jooksul vastavas suunas edasi liikunud. Kui põlengu varjutaja on tekstuuri ära renderdanud, antakse see maapinna varjutajasse ja kujutatakse põlenguna maapinna geometriale. Joonisel 22 on vasakul näha põlengu leviku andmetekstuuri ja paremal genereeritud maastikku. Andmetekstuuril on mustaga kujutatud simulatsioonile ette antud põlengu alguspunkt ja helesinisega on kujutatud põlenud maa ala.

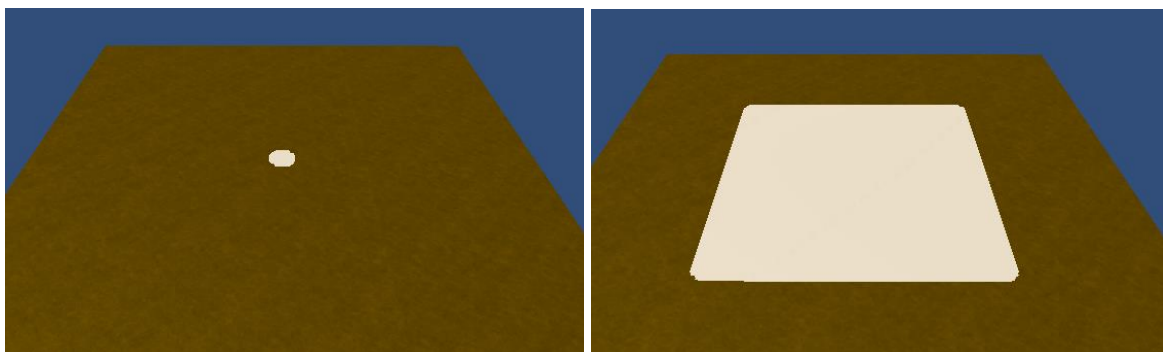


Joonis 22. Põlengu andmetekstuur vasakul, maastikule valgelt kujutatud põleng paremal.

4.3.3 Põlengu sämplimine rakuautomaadi abil

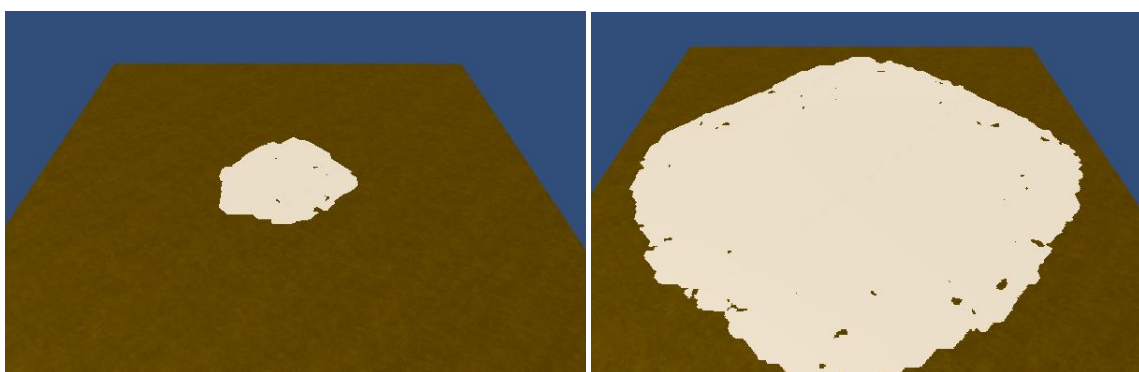
Esimeses rakenduse versioonis kasutati põlengu sämplimiseks rakuautomaati. Rakuautomaat on kombinatsioon värvimata ja värvitud ruutudest, mis areneb vastavalt ette antud reeglitele, naaberrakkude oleku põhjal. Neid reegleid rakendatakse iteratiivselt nii palju kui vajalik. Von Neumann oli üks esimesi, kes sellist mudelit kasutas. Kõige lihtsam variant rakuautomaadist on ühe dimensiooniline joon ning ainult kõrvuti seisvate rakkude arvestamine. Eksisteerib ka kõrgema dimensioonilisi rakuautomaate, mis kasutavad ruudustiku asemel kolmnurki, kuusnurki või kuubikuid [9].

Tulekahju ala hoiustamiseks on kasutusel tekstuur, mille iga piksli punase värvi kanalis on põlengu leviku andmed. Varjutajas vaadatakse iga piksli puhul 3x3 ruutu hetkel arvutatava piksli ümber. Pikslite ortogonaalsest paiknemisest tingituna kujunes põlengu ala ruudu kujuliseks.



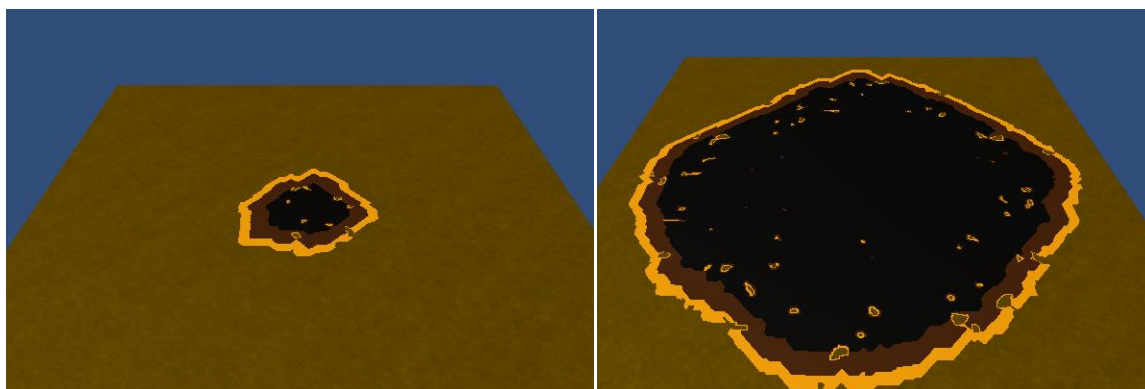
Joonis 23. Ruudus liikumine.

Selle probleemi parandamiseks lisati sämplimisele müra, mis lisab simulatsioonile juhuslikkust ning muudab põlenud ala ümaramaks, nagu on näha jooniselt 24. Sellegipoolest on leviku ala ruudukujulise struktuuriga ja simulatsiooni jaoks sobimatu.



Joonis 24. Müraga liikumine.

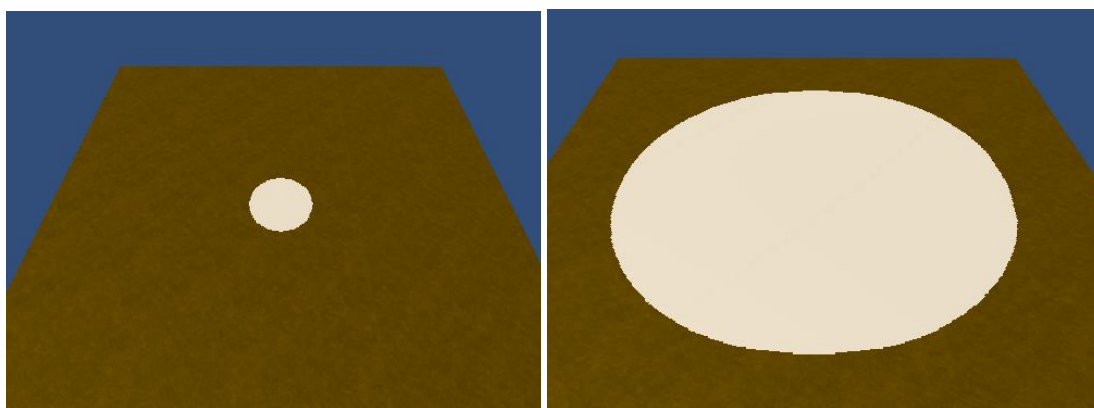
Järgnevalt on joonisel 25 näha sama versioon põleva ala erinevate värvidega, kus must imiteerib põlenud ala, pruun tugevat põlengut ja kollane hiljuti süttinud ala ja tulepiiri.



Joonis 25. Värvidega liikumine.

4.3.4 Ringikujuline sãmplimine

Uue lähenemisena kujutatakse põlengu ala kaugusväljana, kus piksli värv tekstuuril on seoses tema kaugusega tulejoonest. Seda kaugusvälja hoitakse andmetekstuuri punase värvi kanalis, kus tekstuuri värvid on vahemikus 0-1, siis põlenud pikslite väärtus on 0, põlemata pikslitel 1 ning tulejoonel 0.8. Varjutajas vaadatakse läbi iga piksel. Punktide sãmplimist piksli ümbruses tehakse ringikujuliselt. Joonisel 26 on näha põlengu levikut 1200 iteratsiooni tulemusena, kus iga piksli ümber sãmpliti 100 punkti ringis raadiusega 0.01% tekstuuri suurusest.



Joonis 26. Ringis liikumine.

Visuaalselt jäi põlengu ala ringikujuliseks ja laienemine toimus ühtlase kiirusega. Ringi kuju kontrollimiseks viidi läbi test, kus mõõdeti liikumisel esimese saja piksli ja sellele järgneva saja piksli läbimist mööda ringi raadiust. Viie katse tulemusena jääb kiiruse kõikumine alla 1%, mis on simulatsiooni jaoks piisav täpsus.

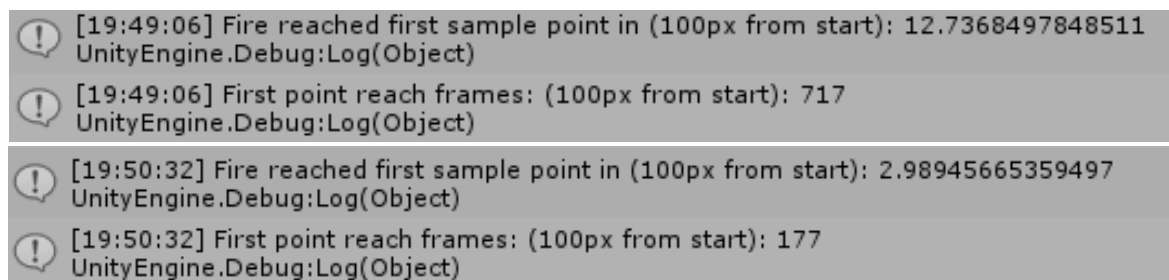
Selleks, et oleks võimalik simulatsiooni liikumist kordaja kaudu aeglasemaks ja kiiremaks muuta, on koostatud valem (1), mis on tuletatud mõõtmistulemustest joonisel 1 (x on kiiruse kordaja). Tänu sellele saab kasutaja soovi korral simulatsiooni kindlel arv kordi kas kiiremaks või aeglasemaks muuta. Sama valemit kasutatakse ka konstantse liikumise saamiseks. See tähendab et iga kaadri puhul, vastavalt sellele kaua kulus eelmise kaadri renderdamiseks, saab varjutaja ette muutuja, millega vajadusel muudetakse järgmise kaadri jaoks põlengu leviku kiirust. Nii on reaajas liikumine eri jõudlusega arvuteid kasutades sama kiire ja ei teki kaadri arvutustele kulunud aja möödumisest tingitud vigu.

Tabel 1. Põlengu leviku kordaja ja kaadrite vaheline seos.

Kordaja	1	2	3	4	5	7	10
Kaadrite arv	717	354	237	180	146	107	80

$$y = 37.74869 + (1466.494 - 37.74869) / (1 + (x / 0.9240277) ^ 1.534914) \quad (1)$$

Valemi saamiseks kasutati MyCurveFit¹ programmi, mis ette antud punktide ja nende väärtuste põhjal teeb matemaatilise valemi. Joonisel 27 on näha kordaja täpsuse testi, kus esimesel juhul on kiiruse kordaja 1, teisel juhul 4. Esimesel juhul on neljandik 179.25, nelja kordse kiirusega läks aega 177 kaadrit, seega on erinevus ~1.2%, millest visuaalse simulatsiooni jaoks piisab.



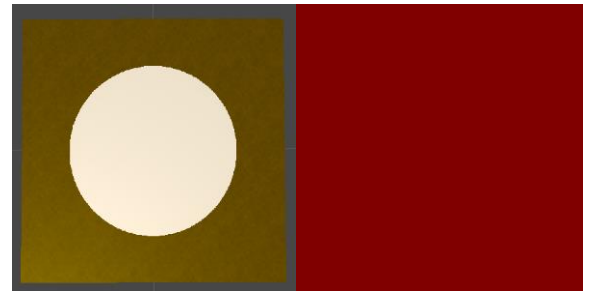
Joonis 27. Põlengu levimis kiiruse kordaja testi tulemus.

¹ <https://mycurvefit.com>

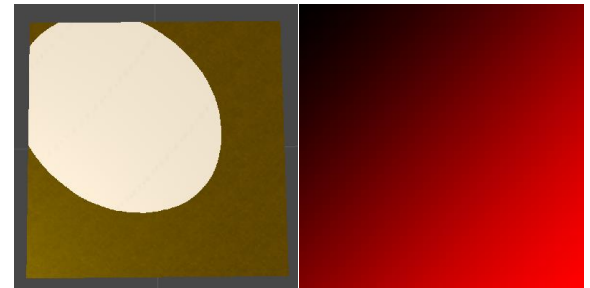
4.4 Välistegurite arvestamine

Simulatsioon toetab temperatuuri ja niiskuse mõju põlengu levikule. Ka nende arvestamiseks on simulatsioonis eraldi andmetekstuurid, millest hetkel on kasutusel punase värvi kanal, teised on vabad hilisemateks täpsustusteks ja lisadeks. Paremalt on näha nii temperatuuri, niiskuse kui ka neutraalse teksturi ette andmisel nähtav tulemus (Andmed on ebarreaalsed ja kasutusel ainult testimise jaoks). Joonisel 28 on põlengu ala neutraalse temperatuuri ja niiskusega, kus vastavad tegurid ei kiirenda ega aeglusta põlengu leviku kiirust. Joonisel 29 on näha temperatuuri ja joonisel 30 niiskuse mõju põlengule, tulenevalt sisendandmetest.

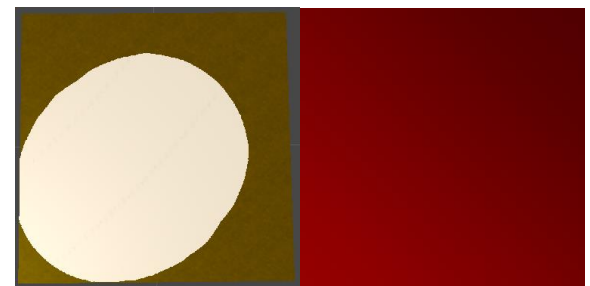
Tuule kiiruse mõju arvutamiseks antakse varjutajale ette tuule kiirus ja vektorkujul tuule suund. Joonisel 31 on vasakul näha põlengu levik ilma tuuleta ja paremal 20km/h tuule korral. Võrdluseks on toodud joonisel 5 võetud 24km/h tuule kiirusega tekkiv põlengu ala kuju. Vastav põlengu kuju saamiseks on leitud samplitava punkti ja tuulesuuna vektori dot produkt, ehk kui vektorid on samasuunalised, on tulemus 1 ja kui vastassuunalised siis -1. Ülejäänud juhtudel jääb arv sinna vahele. Saadud tulemus võetakse astmesse 20, mis annab joonisel 31 nähtava põlengu kuju.



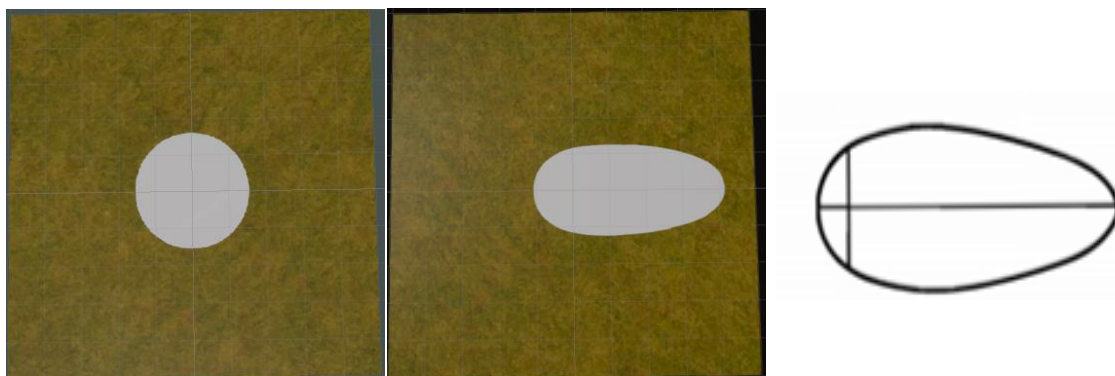
Joonis 28. Põlengu tulemus neutraalse temperatuuriga.



Joonis 29. Temperatuuri mõju põlengule.



Joonis 30. Niiskuse mõju põlengule.



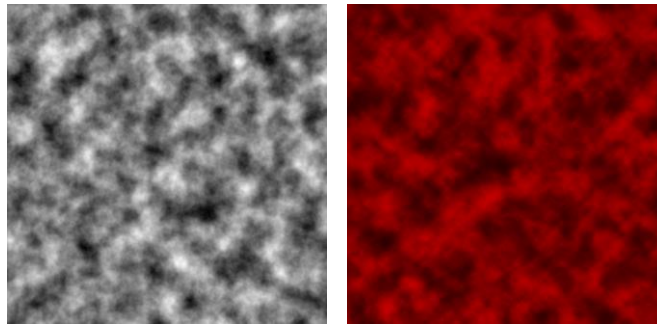
Joonis 31. Tuule mõju põlengule. Vasakul ilma tuuleta, keskel 20km/h tuule kiirusega ja paremal on näide, milline peaks olema põlengu kuju reaalsuses.

Graafikute põhjal tehtud valemid, mille leidmine on täpsemalt välja toodud järgnevas peatükis 5, on loodud samuti kasutades MyCurveFit-i. Vastavalt ette antud tuule kiiruse, tuule suuna, temperatuuri ja niiskuse tekstuurile, leiab varjutaja koefitsiendid, kui palju hetkel vaadeldava piksli värvi vähendada ehk kui palju on tuli vastavale pikslile levinud.

4.4.1 Perlini müra

Perlini müra on gradientmüra, mille lõi Ken Perlin 1983. aastal. Selle algoritmi loomise eest autasustati teda 1995. aastal. Müra loomiseks kasutatakse juhuslikke vektoreid. Perlini müra on laialdaselt kasutusel erinevate arvutigraafika reaalajas toimuvates simulatsioonides [7].

Antud simulatsioonis on kasutatud Perlini müra, et imiteerida niiskuse erinevust looduses. Andmete olemasolul saab kasutada ka kasutaja enda andmetel loodud niiskuse tekstuuri. Joonisel 32 on paremal näha mitme tugevusega kombineeritud müra, mis on simulatsioonis standardina kasutusel.

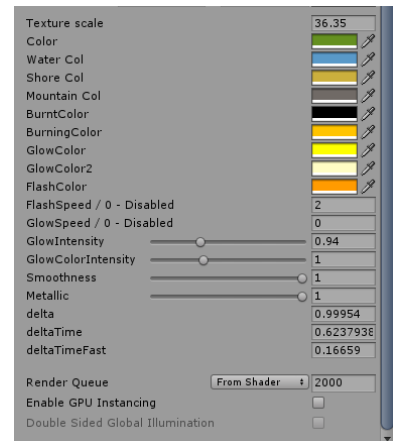


Joonis 32. Perlini müra [8] vasakul, genereeritud niiskuse tekstuur paremal.

4.5 Kasutajaliides ja visuaalne pool

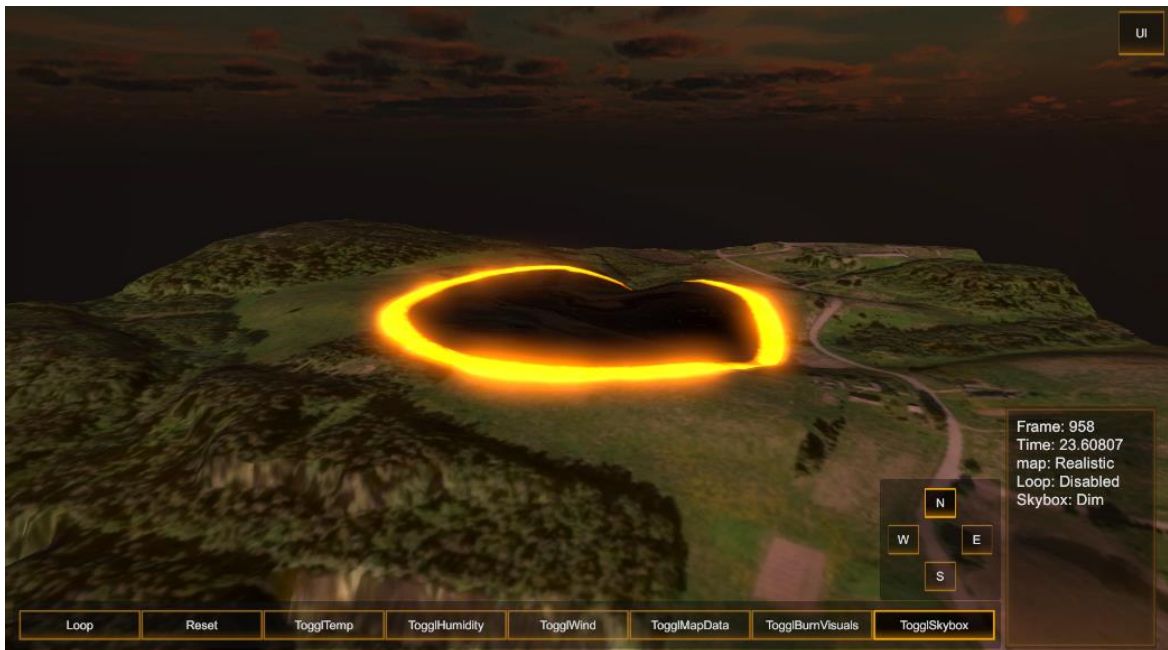
Tule ala jaoks on kasutusel väga lihtne varjutaja, mis vastavalt tule leviku tekstuurile renderdab põlengu maastikule. Võimalik on kasutada erinevaid efekte, mida saab Unity kasutajaliidesest muuta, nagu on näha joonisel 33. Allpool on toodud näide kuidas võiks põlengu ala välja näha.

Lisatud on ka lihtne kasutajaliides (joonis 34), kus saab parameetreid reaalajas muuta või kasutajaliidese üldse ekraanilt ära peita. Ekraanile kuvatakse teavet kaadrite, möödunud aja ja mõningate hetkel aktiivsete parameetrite kohta.



Joonis 33. Põlengu varjutaja efektide muutmise

Võimalik on jätta simulatsioon kordamise peale, kus protseduurilise maastiku puhul genereeritakse igal kahe minuti tagant uus maastik. Selleks saab maastiku kõrguskaarti genereeriv varjutaja ette kaks juhuslikult valitud arvu, mille põhjal genereeritakse kõrguskaardi jaoks vajalikud mürad. Põleng algab alati vastava maastiku keskelt. Perioodiliselt uuesti genereeritud maastik oleks kõige parem näiteks Tartu ülikooli Delta maja ekraanidele visualisatsiooniks.



Joonis 34. Simulatsiooni kasutajaliides.

Selleks et maastiku ümber ei jääks tühjus ja kõik parem välja näeb on lisatud ka erinevad taevakastid. Täpsemalt on need kuubikukaardid, mis määravad kauguses oleva maailma. Tegelikult on tegu lihtsalt omavahel kokku põimitud 6 erinevast pildist, mis moodustavad suure kuubiku.

Visuaali poolest on lisatud kuus erinevat taevakasti, mille vahel saab kasutaja valida. Esimeseks on Unity standard taevakasti ja suunatud valgust kasutades tekitatud öö ja päeva efekt, ning ülejäänud viis on pärit Asset Store'st, mis on üles laetud kasutaja RKD poolt.

4.5.1 Järeltöötlus

Selleks et visuaalset poolt veel ilusamaks saada, kasutatakse tihti erinevaid järeltöötluse efekte. Täpsemalt tähendab järeltöötlus tervele ekraanile erinevate filtrite rakendamist, millega saab imiteerida reaalse kaamera füüsikalisi omadusi. Väheses ajaga saab visuaalset poolt tunduvalt paremaks muuta [23].

Järeltöötluse jaoks on kasutatud Unity Technologies poolt tehtud efektide kogumit [13]. Unity dokumentatsioonis on seda kutsutud über-efektiks, mis kombineerib palju erinevaid efekte üheks suureks kogumiks. Tänu sellele saab ühekorraga rakendada palju erinevaid efekte ja need on alati õiges järjekorras. Nii toimub kõige muutmine ühest kohast, mis on palju kasutajasõbralikum. Veel on vastavasse efektide kogumisse sisse ehitatud moodul efektide silumiseks [14].

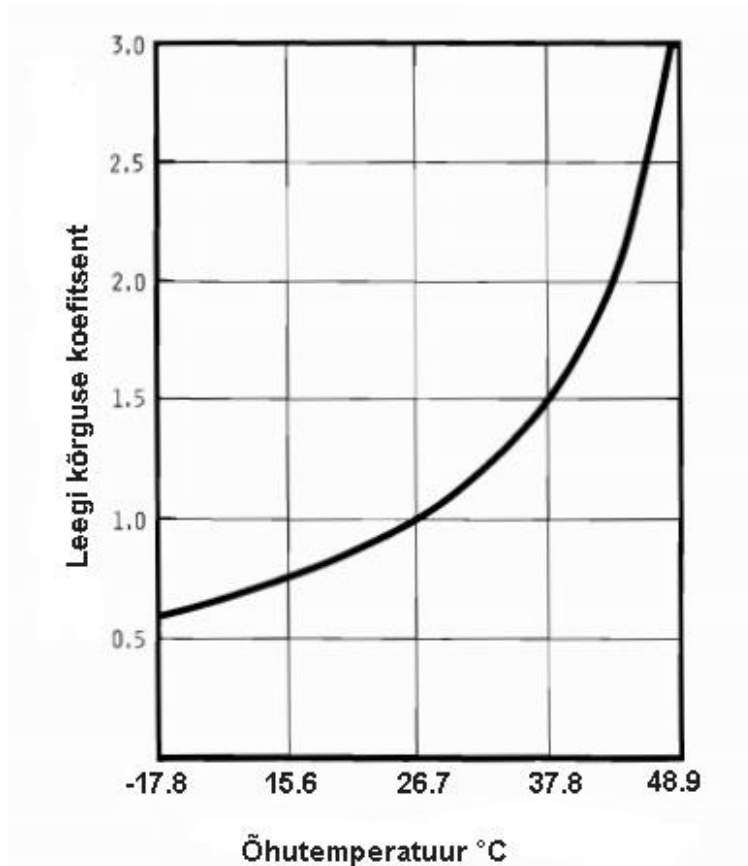
Bakalaureusetöö raames valminud simulatsioonis on järeltöötlusel kasutatud järgmisi efekte:

- Udu – annab kaugusesse tumedama kuma.
- Sakitõrje (*anti-aliasing*) – ühtlustab servasid, vältides teravaid ääri.
- Silma kohanmine (*Eye adaption*) – Mõeldud muidu silma valgusega kohanemise tunde tekitamiseks. Hetkel töötab hästi, kui kasutaja vahetab erinevaid taevakaste ja toimub üleminek hämarast valgesse.
- *Bloom* – Mõeldud realistliku ereda objekti filmimisel tekkiva valguse kuma tekitamiseks. Simulatsioonis kasutusel ereda valgusega põlengu servas valguse kumaks.
- *Vignette* – Tumendab kergelt ekraani ääri, suunates fookuse rohkem ekraani keskele.

5. Simulatsiooni jaoks arendatud valemid

Järgnevalt on välja toodud temperatuuri, niiskuse, topograafia ja tuule kiiruse mõju valemid ja nende tuletamine graafikutest. Joonised on pärit Frank A. Albini uurimusest “Estimating wildfire behaviour and effects” [2]. Valemite loomisel oli abiks MyCurveFit programm [12].

5.1 Temperatuur



Joonis 35. Temperatuuri mõju graafik [2].

Tabel 2. Öhuteratuur ja vastav koefitsient.

0°F	30°F	60°F	70°F	80°F	90°F	100°F	110°F	120°F
-17.8°C	-1.1°C	15.6°C	21.1°C	26.7°C	32.2°C	37.8°C	43.3°C	48.9°C
0.6	0.675	0.75	0.875	1.0	1.25	1.5	2.0	3.0

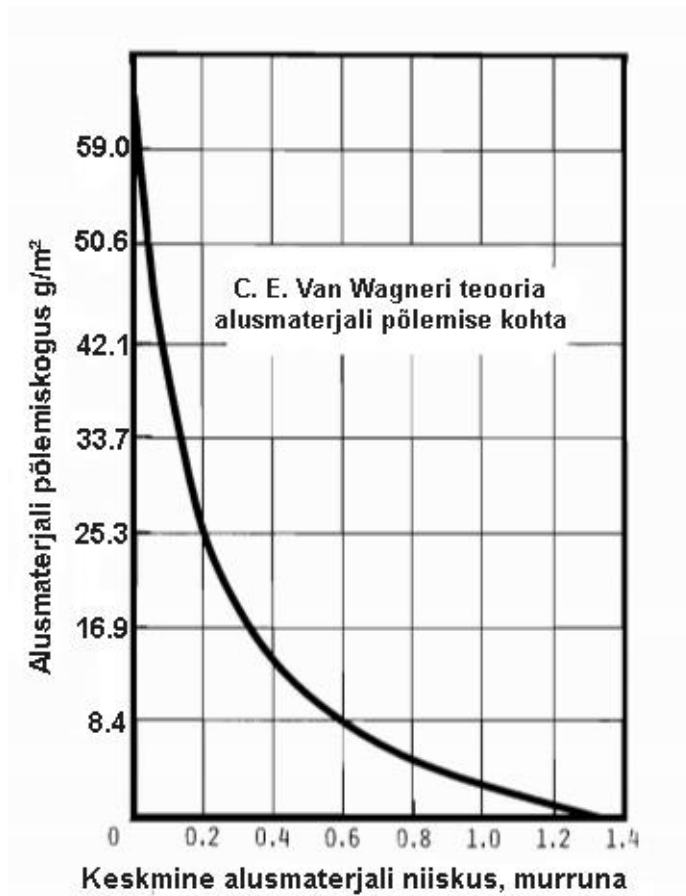
Joonisel 35. oleva graafiku põhjal on koostatud tabel 2, kus on õhuteratuurile vastavad koefitsiendid. Vastava tabeli põhjal on saadud valem 2.

Andmetekstuuril on R-kanalis temperatuur, 0-255, 100 = 0°C, mida on valemi tegemisel arvestatud (temperatuur + 100). Graafikul on tule põletamis kõrgus, kuid simulatsiooni jaoks on seda arvestatud ka levimis kiiruseks temperatuuri mõjul.

Saadud valem (2), kus x on õhuteratuur kraadides ja y on temperatuuri multiplikaator:

$$y = 1746609 + (0.6579554 - 1746609) / (1 + (x / 443.0172) ^ 12.41932) \quad (2)$$

5.2 Niiskus



Joonis 36. Niiskuse mõju graafik [2].

Tabel 3 Niiskus, põlemiskiirus ja vastav kordaja.

Niiskus	0	0.1	0.2	0.4	0.6	0.8	1.0	1.2
Põlemiskiirus	65	42.1	25.3	14	8.4	4.2	2.8	1.68
Kordaja	2.569	1.664	1.0	0.553	0.332	0.166	0.111	0.066

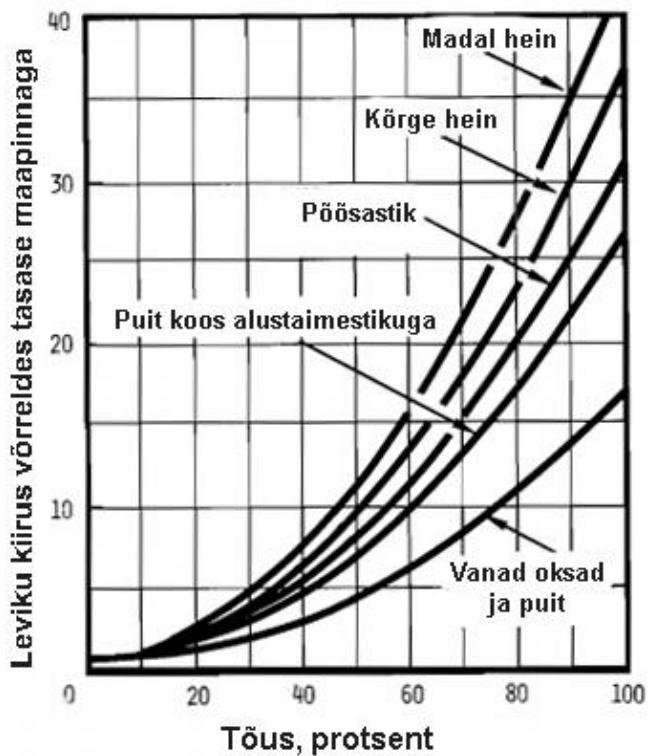
Joonisel 36. oleva graafiku põhjal on koostatud tabel 3, kus on niiskusele vastavad põlemiskiirused ja kordajad. Vastava tabeli põhjal on saadud valem 3. Simulatsiooni jaoks on tavaväärtuseks, mille juures jääb kordaja 1, võetud 25.3g/m² kohta.

Andmetekstuuril on R-kanalis niiskus, 0-1. Punasesse kanalisse läheb niiskus korda 0.714276 et saada väärtust 0-st kuni 1ni. Graafikul on keskmine alusmaterjali niiskus, kuid simulatsiooni jaoks on seda arvestatud tule levimiskiiruseks niiskuse mõjul.

Saadud valem (3), kus x on niiskus ja y on niiskuse multiplikaator:

$$y = -0.1152386 + 2.6875636 / (1 + (x / 0.1626785)^{1.297992}) \quad (3)$$

5.3 Topograafia



Joonis 37. Topograafia mõju graafik [2].

Tabel 4. Tõusule vastav kordaja.

Protsent	0	20	40	50	60	70	80	90	100
Kordaja	1	2.5	5	6.5	10	12.5	17.5	22.5	26

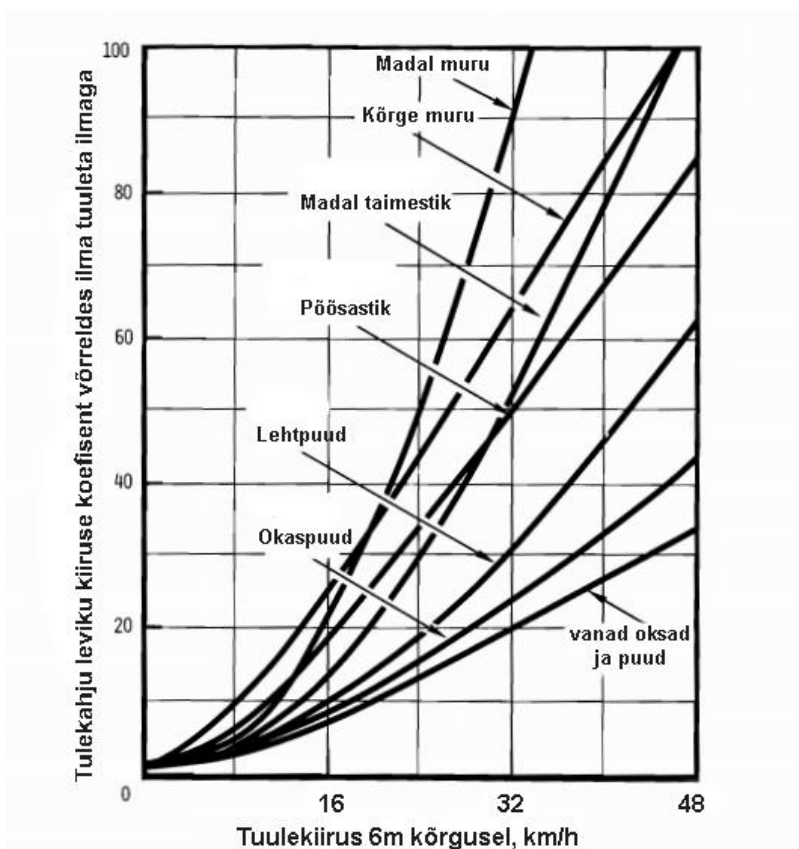
Joonisel 37. oleva graafiku põhjal on koostatud tabel 4, kus on õhutemperatuurile vastavad koefitsiendid. Vastava tabeli põhjal on saadud valem 4.

Andmetekstuuril on igas värvi kanalis vastava piksli kohta info igas neljas suunas liikumise kohta, kui palju on tõusu või langust. Tõusu puhul on arv 0 – 0.5, languse puhul 0.5 – 1. Protsendi saamiseks tõusu puhul: tõus * 2, languse puhul: (langus – 0.5) * 2.

Saadud valem (4), kus x on tõus protsendina ja y on tõusu/languse multiplikaator:

$$y = 97.41277 + (1.485698 - 97.41277) / (1 + (x / 150.3879) ^ 2.566629) \quad (4)$$

5.4 Tuul



Joonis 38. Tuule kiiruse mõju graafik [2].

Tabel 5. Tuule kiirus ja sellele vastav kordaja.

Tuule kiirus	0	8	16	24	32	40	48
Kordaja	0	5	10	20	30	45	63

Joonisel 38. oleva graafiku põhjal on koostatud tabel 5, kus on õhutemperatuurile vastavad koefitsiendid. Vastava tabeli põhjal on saadud valem 5.

Varjutaja saab sisendiks tuule kiiruse ja selle suuna.

Saadud valem (5), kus x on tuule kiirus 6m kõrgusel ja y on tuule kiiruse multiplikaator:

$$y = 50749390 + (1.175058 - 50749390) / (1 + (x / 106128.2)^{1.769543}) \quad (5)$$

6. Täpsus, testimine ja edasiarendused

Töös loodud simulatsioon põhineb Rothermeli peamiselt labori katsete põhjal koostatud mudelitel. Samas ei ole neis mudelites arvestatud kõigi looduse eripäradega. Näiteks põleva materjali ebahühtlane jagunemine. Samuti pole täielikult testitud mitme erineva teguri koosmõju täielikult testitud. Teiste samuti Rothermeli mudeli põhinevate simulatsioonide võrdlemisel reaalsete metsatulekahjudega on erinevus jäänud keskmiselt paarikümne protsendi juurde [2].

Peamine viga tekib üldjuhul põleva materjali ebahühtlasest jagunemisest, tuule kiiruse erinevustest, tõusust ja langusest ning niiskuse kõikumisest. Vastavaid parameetreid ei ole võimalik saja protsendilise täpsusega mõõta, kuna looduses ideaalseid ja konstantseid olusid ei eksisteeri [2].

Antud mudelite ja mõõtmistulemustest tingitud vigade suurusjärgu tõttu, jääb simulatsiooni enda veaprotsent pea olematuks. Arvestades mudelite paarikümne protsendilist kõikumist, siis pole simulatsiooni jaoks arendatud valemite ümardusvead oma suurusjärgu tõttu niivõrd olulised.

Käesolevas töös on puudu veel mõningad Rothermeli mudelis arvestatud tegurid nagu maastiku tõus ja langus ning põleva materjali tüüp, mis jäi käesoleva töö skoobist välja. Simulatsioonis arvestatakse materjali niiskust, õhutemperatuuri ja tuult, mis annavad juba hea ülevaate, kuidas keskkonnatingimused põlengu levikut mõjutavad.

6.1 Jõudlus

Siin peatükis antakse ülevaade simulatsiooni soovitatavate nõuete ja katsetuste kohta. Testimine on läbi viidud ühel laua- ja sülearvutil.

Lauaarvuti konfiguratsioon on järgmine: protsessor: I5-3570K, 4.0GHz, operatiivmälu: 16GB, videokaart: GTX660 2GB, kõvaketta tüüp: SSD, ekraani resolutsioon: 1920 x 1080.

Testimise tulemusena oli keskmine kaadrisagedus 60 kaadrit sekundis. Programmi töös ei täheldatud anomaaliaid ning simulatsioon oli täielikult sujuv. Mälukasutus ei ületanud 200MB ja protsessori hõivatus jäi peale maastiku genereerimist alla 12%.

Sülearvuti konfiguratsioon on järgmine: protsessor: I5-3230M, 2.6GHz, operatiivmälu: 6GB, videokaart: GTX635M 2GB, kõvaketta tüüp: SSD, ekraani resolutsioon: 1366 x 768.

Testimise tulemusel oli keskmine kaadrisagedus 40 kaadrit sekundis. Programmi töös ei täheldatud anomaaliaid ning simulatsioon oli enamjaolt sujuv. Esinesid üksikud kaadrisageduse langused. Mälukasutus ei ületanud 300MB ja protsessori hõivatus jäi peale maastiku genereerimist alla 20%.

7. Kokkuvõte

Käesolevas bakalaureusetöös kirjeldati metsatulekahjusid ja nende levikut. Uuriti erinevaid põlengu leviku simulatsiooni programme ja tehti levinumatest ülevaade. Töö raames implementeeriti kaasaegne ja silmatorkav metsatulekahjude simulatsioon, mis on visuaalne näide sellest, kui ohtlikult ja kiiresti võivad metsatulekahjud levida ning kui suur mõju on soodsatel teguritel tulekahju levikule. Praegu sobib see heaks visualisatsiooniks nii lastele kui ka täiskasvanutele näitamaks, et tulega tuleb ettevaatlikult ümber käia ja kuivadel perioodidel hooletu kulu põletamine võib väga kiiresti kontrolli alt väljuda.

Rakendus valmis mängu-mootoris Unity. Maastiku genereerimiseks on võimalik kasutada laserskanneerimisandmeid, millele lisades satelliitfoto, saab näha põlengut levikut päris maastikul. Maastikku saab ka protseduuriliselt genereerida.

Simulatsioonis kasutatud valemid on koostatud Frank A. Albini poolt loodud graafikute põhjal, mis põhinevad omakorda Rothermel'i mudelitel. Põlengu levikut arvutatakse varjutajas, kus nii sisend- kui ka väljundandmed saadetakse andmetekstuuridena. Põlengu andmetekstuuri pikslite arvutamiseks on kasutatud ringis punktide sümplimist. Laserskanneerimise andmed töödeldakse enne kasutamist, eemaldades mõõtmisel tekkinud vead. Selleks vaadeldakse iga skanneeritud punkti ümbrust spiraalse sümplimise kaudu ning viigased punktid jäetakse arvestamata.

Valminud simulatsioon on kättesaadav rakenduse kujul töö lisas koos lihtsa kasutajaliidese ja lisatud järeltöötlus efektidega. Selles saab reaajas muuta erinevaid tegureid ja näha, mis mõju on põlengule näiteks ilmastikuoludel. Rakenduses on toetatud ka kordamisfunktsioon, kus iga kahe minuti möödudes genereeritakse protseduuriliselt uus juhuslik maastik ning põleng algab uuesti. See sobiks hästi näiteks Tartu Ülikooli Delta maja ekraanidele visualisatsiooniks.

Hetkel pole simulatsioon veel realistlike olukordade läbimängimiseks valmis, kuna mõnede oluliste parameetritega arvestamine, nagu taimestiku liik ja tihedus jäi antud töö skoobist välja. Kui valemid ja loogika veel täpsemaks muuta, sobib rakendus reaalses olukorras kasutamiseks põlengute kustutamise planeerimisel või riskianalüüside tegemisel. Täpsemalt oleks vaja uurida erinevate tegurite koosmõju ning võimalusel kasutada kaasaegsemaid tulekahju leviku valemiteid ja mudeleid. Lisaks võib implementeerida võimaluse arvutada kustutamiseks vajamineva inimressursi ja tehnika suurusjärku. Teiseks ideeks on luua lastele suunatud arvutimäng, kus peab kasutama strateegilist mõtlemist ja saadaolevaid ressursse ära kasutades tulekahju võimalikult kiirelt kustutama. Vastav mäng oleks huvitav ja kaasahaarav, kuid ka väga õpetlik näitamaks, kui kiiresti võib kontrolli alt väljunud lõke suureks põlenguks muutuda.

8. Viidatud kirjandus

- [1] Forest Fires. https://web.archive.org/web/20151016185535/http://www.bbc.co.uk/science/earth/natural_disasters/forest_fire (12.01.2019)
- [2] Frank A. Albini. Estimating Wildfire Behavior and Effects. INTERMOUNTAIN FOREST AND RANGE EXPERIMENT STATION Forest Service U.S. Department of Agriculture Ogden, 1976.
- [3] Is Global Warming Fueling Increased Wildfire Risks. <https://www.ucsusa.org/global-warming/science-and-impacts/impacts/global-warming-and-wildfire.html> (13.01.2019)
- [4] Bonsor K. How Wildfires Work. <https://science.howstuffworks.com/nature/natural-disasters/wildfire.htm>
- [5] Talvari A. HÄDAOLUKORRAD EESTIS: merereostus ja metsatulekahjud. Sisekaitseakadeemia, 2011.
- [6] Siegel E. The Terrifying Physics of How Wildfires Spread So Fast. <https://www.forbes.com/sites/startswithabang/2017/09/06/the-terrifying-physics-of-how-wildfires-spread-so-fast/#4c4df0107791>
- [7] Perlin noise. https://en.wikipedia.org/wiki/Perlin_noise (25.03.2019)
- [8] Scher Y. Playing with Perlin Noise: Generating Realistic Archipelagos. <https://medium.com/@yvanschery/playing-with-perlin-noise-generating-realistic-archipelagos-b59f004d8401> (25.03.2019)
- [9] Cellular Automaton. <http://mathworld.wolfram.com/CellularAutomaton.html> (25.03.2019)
- [10] Maa-amet. Eesti topograafia andmekogu ruumiandmed. https://geoportaal.maaamet.ee/index.php?lang_id=1&page_id=607#tab3 (28.03.2019)
- [11] Draw equidistant points on a spiral. <https://stackoverflow.com/questions/13894715/draw-equidistant-points-on-a-spiral> (26.04.2019)
- [12] MyCurveFit. <https://mycurvefit.com/> (28.04.2019)
- [13] Unity, Post Processing Stack. <https://assetstore.unity.com/packages/essentials/post-processing-stack-83912> (28.04.2019)
- [14] Unity dokumentatsioon, Post-processing stack. <https://docs.unity3d.com/Manual/PostProcessing-Stack.html> (24.04.2019)
- [15] Wildfireanalyst. <https://www.wildfireanalyst.com/features/> (28.04.2019)
- [16] FARSITE. <https://www.firelab.org/project/farsite> (29.04.2019)
- [17] Shader. <https://en.wikipedia.org/wiki/Shader> (29.04.2019)
- [18] GPU vs CPU Computing: What to choose. <https://medium.com/altumea/gpu-vs-cpu-computing-what-to-choose-a9788a2370c4> (29.04.2019)

[19] Simon M. These wind patterns explain why california's wildfires are so bad <https://wildfiretoday.com/2018/06/07/engineer-who-worked-on-plans-for-nuclear-powered-airplane-later-developed-the-fire-spread-model/> (03.05.2019)

[20] NWSBayArea. https://twitter.com/NWSBayArea/status/1061657541501960195/photo/1?ref_src=twsrc%5Etfw%7Ctwcamp%5Etweetembed%7Cwt-term%5E1061657541501960195&ref_url=https%3A%2F%2Fwww.wired.com%2Fstory%2Fthese-wind-patterns-explain-why-californias-wildfires-are-so-bad%2F (03.05.2019)

[21] Gabbert B. [Engineer who worked on plans for nuclear-powered airplane later developed the fire spread model.](https://wildfiretoday.com/tag/dick-rothermel/) <https://wildfiretoday.com/tag/dick-rothermel/> (03.05.2019)

[22] Fotini-Niovi Pavlidou ja George D. Papadopoulos. Software tools for wildfire monitoring. https://www.researchgate.net/publication/267798936_SOFTWARE_TOOLS_FOR_WILDFIRE_MONITORING (03.05.2019)

[23] Unity dokumentatsioon, Post-processing overview. <https://docs.unity3d.com/Manual/PostProcessingOverview.html> (06.04.2019)

Lisad

I. Mõisted ja terminid

Unity on Unity Technologies poolt arendatud mängumootor, mis tuli välja 2005 aasta juunis. Nüüdseks toetab enam kui 27 erinevat platvormi. Tuntumad neist on Windows, Linux, Android, PS4 ja XBOX ONE. Mängumootorit saab kasutada nii kahe- kui kolme-dimensiooniliste mängude ja simulatsioonide tegemiseks. Unity puhul on peamiseks programmeerimis keeleks C#. ¹

Varjutaja (ingl. k *shader*) on arvutigraafikas kasutusel olev termin. Varjutaja tegeleb erinevate efektide renderimisega, mida põhiliselt tehakse arvuti graafikakaardis. Täpsemalt määrab varjutaja kas siis piksli või tipu omadusi. ²

Kõrguskaart (ingl. k *heightmap*) on 3D arvutigraafikas kasutusel olev termin, mida kasutatakse raster pildi kohta. Iga piksli alla salvestatakse vajalikud andmed, näiteks maastiku kõrgus. Heightmap'e võib visuaalselt kujutada mustavalge pildina, kus must värv märgib miiminum kõrgust ja valge maksimum kõrgust. Vastava pildi põhjal saab genereerida arvutimängu või simulatsiooni maastiku. ³

GameObject on põhiline Unity objekt, mille alla kuuluvad kõik võimalikud esemed, karakterid ja dekoratsioonid. Gameobject'id on peamiselt konteinerid, mis küll ise suurt midagi ei tee, vaid hoiavad endas komponente, millest tuleb funktsionaalsus. ⁴ Unity dokumentatsioonis on toodud näiteks Valgusobjekt, mis on tühi GameObject, kuhu on külge lisatud valguse komponent.

Kuubikukaart (ingl. k *cubemap*) on kuue ruudu kujulise tekstuuri kollektsioon, mis kujutab endast ümbruse peegeldust. Tegu on kujutletava kuubikuga, mis ümbritseb objekti, iga külg on vaade ühes suunas. ⁵

Peegeldussond (ingl. k *reflection probe*) on umbes nagu kaamera, mis näeb kera kujuliselt igas suunas enda ümbrusesse. Vastav pilt salvestatakse kuubikukaarti, mida saab kasutada peegeldavate objektide peegelduse saamiseks. Tseenis võib kasutada mitmeid peegeldussonde ja panna objekte kasutama endale lähimat. ⁶

¹ [https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine))

² <https://en.wikipedia.org/wiki/Shader>

³ <https://en.wikipedia.org/wiki/Heightmap>

⁴ <https://docs.unity3d.com/Manual/class-GameObject.html>

⁵ <https://docs.unity3d.com/Manual/class-Cubemap.html>

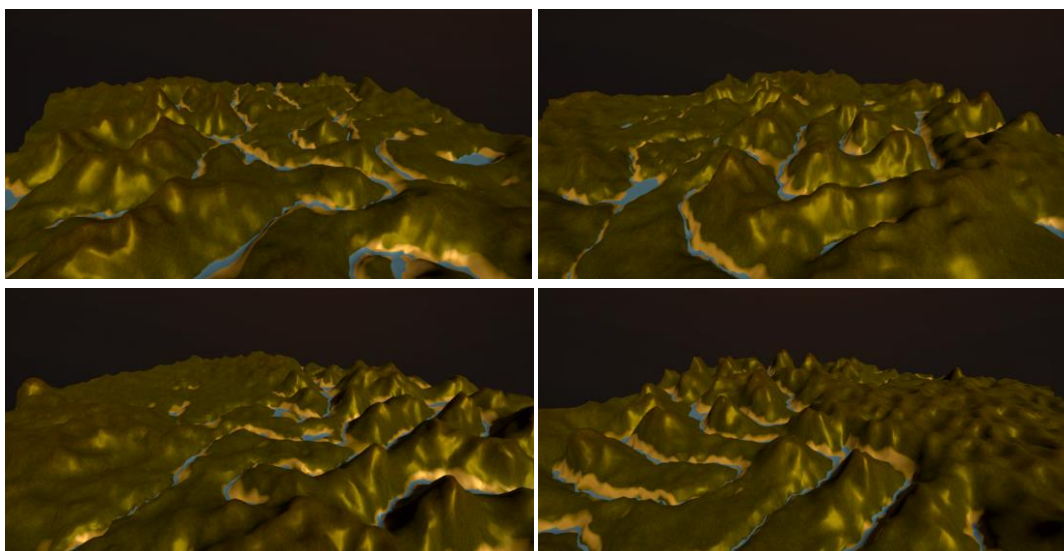
⁶ <https://docs.unity3d.com/Manual/class-ReflectionProbe.html>

II. Rakenduse pildid

Siin on näha simulatsioonis kasutatud taevakastid erinevates ilmastikuoludes.



Järgnevalt on välja toodud mõningad protseduuriliselt genereeritud maastikud.



III. Rakenduse kasutamine

Simulatsiooni rakendus on tööga kaasas olevas zip-failis “Metsatulekahjude_Simulatsioon_Reaalajas.zip”. Käivitamiseks tuleb vastav fail lahti pakkida ja avada programm “Metsatulekahjude_simulatsioon.exe”.

Rakenduse klahvide kirjeldused:

- A – liigutab kaamerat vasakule.
- D – liigutab kaamerat paremale.
- S – liigutab kaamerat alla.
- W – liigutab kaamerat üles.
- Z – liigutab kaamerat keskpunktist kaugemale.
- X – liigutab kaamerat keskpunktile lähemale.
- P – peatab põlengu leviku ja aja möödumise.
- Escape – sulgeb rakenduse.

IV. Lähtekood

Lähtekood on saadaval GitHubis: https://github.com/ilysion/Metsatulekahjude_simulatsioon_reaalajas

Lähtekoodi jaoks on vaja Unity versiooni 2018.3.0f2, millega lähtekood on loodud ja kindlasti töötab. Peaks töötama ka uuemate versioonidega, kuid võib esineda mõningaid probleeme.

Peamised materjalid ja varjutajad asuvad Assets/Materials ja selle alamkaustades. Täpsemalt on alamkaustas BurnRelated peamised kasutusel olevad varjutajad. LIDAR kaustast leiab laserskanneerimisega seotud materjalid ja varjutajad. MapRelated kaustas asuvad maastiku geomeetria materjalid. TexturePictures kaustas asuvad andmetekstuurid. Skriptid on Assets/Scripts kaustas ja selle alamkaustas.

Rakenduse kaustastruktuur on järgmine:

- Assets
 - Materials
 - BurnRelated – peamised kasutusel olevad varjutajad
 - LIDAR – laserskanneerimise materjalid ja varjutajad
 - MapRelated – maastiku geomeetria materjalid
 - TexturePictures - andmetekstuurid
 - Scripts – simulatsioonis kasutatavad skriptid
 - Sky5X – kasutusel olevad taevakastid
 - Textures – protseduurilise maastiku tekstuurid

V. Litsents

Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks.

Mina, **Hannes Saariste**,

1. Annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) enda loodud teose
Metsatulekahjude simulatsioon reaajas,

Mille juhendaja on **Jaanus Jaggo**,

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commons'i litsentsiga CC BY NC ND 3.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.
3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.
4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Hannes Saariste

05.04.2019