Autodesk Maya-Efekt lomljenja

Biskupović, Tanja

Undergraduate thesis / Završni rad

2015

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University North / Sveučilište Sjever**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:122:756091

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-03-29**

Repository / Repozitorij:

University North Digital Repository

Sveučilište
Sjever

Završni rad br. 304/MM/2013

AUTODESK MAYA – efekt lomljenja

Tanja Biskupović, 2384/601

Varaždin, rujan 2015. godine

Sveučilište
Sjever

Odjel za multimediju, oblikovanje i primjenu

Završni rad br. 304/MM/2013

AUTODESK MAYA – efekt lomljenja

Student

Tanja Biskupović, 2384/601

Mentor

pred. Andrija Bernik, dipl.inf.

Varaždin, rujan 2015. godine

SAŽETAK

U posljednjih 20-ak godina računalna grafika doživjela je neviđeni procvat. Sve brži razvoj hardverskih rješenja doveo je i do rapidnog razvoja naprednih 3D tehnologija. Uz napredak računalne 3D interakcije, došao je i problem 3D destrukcije. Tu počinje razvoj raznih naprednih sustava za računalnu simulaciju loma, uništenja, destrukcije općenito. Pitanje loma rješava se na razne načine, oni se razlikuju od sustava do sustava, a danas se nudi pregršt raznih alata. Iako se pitanje loma uglavnom veže uz snažne dinamičke sustave koji se bave krutim tijelima, i 3D računalni programi namijenjeni za modeliranje i animaciju idu u korak s tehnologijom te uvode razne alate namijenjene kreiranju efekta loma. Među njima je i program Autodesk Maya koji, uz brojne plug-inove i programske skripte, nudi i vlastiti ugrađeni alat za efekt loma.

Ključne riječi:

3D, Autodesk Maya, destrukcija, dinamika, efekt loma, kolizija, kruta tijela

ABSTRACT

In last two decades computer graphics has undergone the unseen boom. The fast development of new hardware solutions has led to the rapid development of advanced 3D technologies. With the advancement of computer 3D interaction, both came the problem of 3D destruction. That is the development start point for variety of advanced systems for computer simulation of fracture, shatter or destruction in general. The problem of shatter is dealt with in different ways which vary from system to system, and nowadays a variety of tools is offered. Although the question of shatter effect is usually associated with robust dynamic systems build for rigid body simulations, 3D computer programs designed for modeling and animation both keep up with technology, thus introducing a variety of tools for creating the shatter effect. Among them is Autodesk Maya who, besides numerous plug-ins and scripts, has its own built-in tool for shatter effect.

Keywords:

3D, Autodesk Maya, Collision, Destruction, Dynamics, Rigid Body, Shatter Effect

SADRŽAJ

1. UVOD 1

2. GAME ENGINES – POVIJEST DESTRUKCIJE 3

2.1. GeoMod 5
2.2. Frostbite Engine 7
2.3. CryENGINE 11
2.4. Unreal Engines 13

3. TEORIJA LOMA 14

3.1. Vrste destrukcijskih procedura 17
3.1.1. Osnovni RBS pristup 18
3.1.1.1. Lomljenje objekta – priprema geometrije 18
3.1.1.2. Definiranje ograničenja i dizajniranje koreografije 21
3.1.1.3. Pokretanje simulacije i rješavanje detekcije kolizije 21
3.1.2. RBS i specijalizirane fizikalne knjižnice 22
3.1.3. Analiza konačnih elemenata (FEA) 23

4. NAJPOPULARNIJI 3D ALATI 24

4.1. Houdini 24
4.2. 3ds Max 27
4.3. Blender 29

5. AUTODESK MAYA 30

5.1. Dinamička polja 30
5.1.1. Klase 31
5.1.2. Vrste 31
5.2. Efekt lomljenja 33
5.2.1. Lom površine 33
5.2.2. Lom punog tijela 36
5.2.3. Napuknuće 37
5.3. nCloth lomljenje 38
5.4. Plug-inovi za destrukciju 39
5.4.1. Bullet Physics 39
5.4.2. DMM 41
5.4.3. Fracture FX 44
5.4.4. PullDownIt 45
5.4.5. Skripte 48
5.4.6. Usporedba 49
5.5. Utjecaj na performanse 50

6. PRIMJENA – IZRADA VIDEOANIMACIJE 53

6.1. Odabir alata 54
6.2. Scenografija 57
6.3. Lomljenje pomoću ugrađenog alata 58
6.4. Lom pomoću nClotha 64
6.5. Lom pomoću Bullet plug-ina 66
6.6. Lom pomoću PullDownIt plug-ina 68

7. ZAKLJUČAK 71

8. LITERATURA 73

Popis korištenih kratica

API	*Application Programming Interface*	sučelje za programiranje aplikacija
CGI	*Computer Generated Image*	računalno generirana slika
CSG	*Constructive Solid Geometry*	konstruktivna geometrija punih tijela
DMM	*Digital Molecular Matter*	digitalna molekularna materija
Dneg	*Double Negative*	(tvrtka, grafički studio)
FEA	*Finite Element Analysis*	analiza konačnih elemenata
FEM	*Finite Element Method*	metoda konačnih elemenata
FX	*[Digital]Effects*	specijalni/digitalni efekti
ILM	*Industrial Light & Magic*	(tvrtka, grafički studio)
MMO	*Massively Multiplayer Online [Game]*	mrežna [igra] s masovnim brojem igrača
MPC	*Moving Picture Company*	(tvrtka, grafički studio)
PC	*Personal Computer*	osobno računalo
RBD	*Rigid Body Dynamics*	dinamika krutih tijela
RBS	*Rigid Body Simulation*	simulacija krutih tijela
san.	*Sanskrt*	(jezik)
VFX	*Visual [Digital]Effects*	vizualni specijalni/digitalni efekti

1. UVOD

Problem lomljenja, uništavanja, destrukcije općenito sveprisutan je u grafičkoj industriji. Potrebu za simulacijom destrukcije nalazimo u automobilskoj industriji, medicini, sigurnosnim sustavima, elektranama, nuklearnim postrojenjima te, dakako, u filmskoj industriji i industriji računalnih igara. Tradicionalan pristup lomu 3D objekata jest ručno odvajanje na fragmente i animiranje svakog pojedinog fragmenta ključnim okvirima. Uz dobro poznavanje fizike loma, mnogo ručnog rada i kvalitetne grafičare, tim je putem moguće dobiti zadovoljavajuće rezultate – izgledom i realnošću loma – no danas je potrebno više. Uz potrebe za sve kraćim vremenom izrade specijalnih efekata za filmove i rapidno rastuću industriju videoigara te njezinu potrebu za grafikom u stvarnom vremenu, kreiranje loma animacijom ključnim okvirima postalo je nemoguće. Problem destrukcije u stvarnom vremenu postao je glavni pokretač razvoja novih alata za simulaciju loma, a uz rame s njime išao je i problem masivnih razaranja u suvremenim filmovima. Iako se pitanje loma uglavnom veže uz snažne dinamičke sustave koji se bave krutim tijelima, i 3D računalni programi namijenjeni za modeliranje i animaciju idu u korak s tehnologijom te uvode razne alate namijenjene kreiranju efekta loma. Među njima je i program Autodesk Maya koji uz brojne plug-inove i programske skripte, nudi i vlastiti ugrađeni alat za efekt loma.
Ovaj se rad bavi svim spomenutim temama i pitanjima, te preciznije određuje kako se u tome svijetu snašao 3D program Autodesk Maya.

Rad je koncipiran u pet sastavnih cjelina. U prvoj se cjelini daje uvid u problem računalne grafike u stvarnom vremenu, uz naglasak na pitanje destrukcije. Daje se pregled glavnih prekretnica u tome svijetu, oprimjerenih raznim popularnim računalnim igrama. Opisani su sustavi koji su izgradili važne stepenice u povijesti razvoja računalne destrukcije, a koji i danas imaju veliku ulogu na tome području. Svaki je alat objašnjen i opisan te je dan osnovni uvid u njegov način rada.

Druga se cjelina bavi općom teorijom loma: donose se različiti načini računalnog loma uz sažeti uvid u njihov način rada i opću primjenu. Ova je cjelina neobično bitna s obzirom da daje teorijsku potporu i velik dio terminološkog aparata za čitav rad.

Treća cjelina govori kako je pitanje loma riješeno u nekim od najpopularnijih 3D programa, a četvrta se cjelina dovezuje na to i daje detaljan uvid u pitanje loma kod programa

Autodesk Maya. Ova je cjelina utemeljena kako na teorijskoj literaturi tako i na vlastitim praktičnim saznanjima. Detaljno se bavi područjem dinamike, odnosno dinamičkim poljima i ugrađenim alatom za efekt loma. Objašnjavaju se alternativni načini za kreiranje loma te se opisuju najbolji i(li) najpopularniji plug-inovi namijenjeni za destrukciju. Ova se cjelina zaključuje opisom utjecaja na performanse koji je također temeljen i na teorijskim saznanjima i na vlastitim spoznajama dobivenim pri izradi videoanimacije.

Posljednja, peta cjelina, ukratko prolazi proces izrade videoanimacije i tumači neke od problema do kojih je došlo tijekom rada te nudi načine na koje su oni riješeni.

Uz svaku sliku ili prikaz naveden je izvor, a ako nema naznačenog izvora, slika je vlastita. Navođenje izvora u čitavu je radu temeljito i kontinuirano, jednako je i kod citata, ali oni se navode prevedeni na hrvatski jezik. Eventualne intervencije u citatima navedene su u uglatim zagradama. Čitav popis literature dan je na kraju rada.

Pitanje loma u mnogočemu je više fizikalno-matematičko pitanje no grafičko. Osnovna problematika svodi se na nekoliko temeljnih točaka i glavno pitanje: kako lom učiniti što realnijim uz što kraće vrijeme softverske obrade? Vrijeme se stavlja u prvi plan i često postaje presudnim faktorom, kada je u pitanju grafika u stvarnom vremenu, nekada je presuđivalo i nauštrb realističnosti. U svijetu kojemu je glavni stroj sat, a stotinke će mu uskoro biti sastavni dio – takva situacija ne začuđuje. Danas je, ipak, sve veća potražnja za sustavima koji nude sve: besprijekornu realističnost i vrijeme odaziva u razni stvarnog vremena, a najnapredniji sustavi koje odlikuje sveobuhvatnost i nevjerojatna brzina su tzv. *game engini*, programski sustavi za razvoj videoigara.

2. Game engines – povijest destrukcije

U posljednjih 20-ak godina računalna grafika doživjela je procvat. Sve brži razvoj hardverskih rješenja doveo je i do rapidnog razvoja naprednih 3D tehnologija čija je primjena postala posve nezaobilazna: od filmske i automobilske industrije, sve do medicine i astronomije. No, nesumnjivo, računalna grafika najviše zarade donosi industriji računalnih igara. S obzirom da je tek u posljednjem desetljeću dobila ozbiljnije priznanje, te uzevši u obzir kako je zadnjih nekoliko godina industrija računalnih igara zaradila više od filmske i glazbene industrije zajedno [1], logično je zaključiti da će se u nju i sve više ulagati. Od razonode koja je nekada bila u rukama nezavisnih entuzijasta, izrasla je u industriju pod okriljem korporacijskih divova. Povijest razvoja računalnih igara zapravo je povijest razvoja računalne grafike, sve do točke gdje smo danas – računalne 3D grafike u stvarnom vremenu (eng. *real-time 3D graphic*).

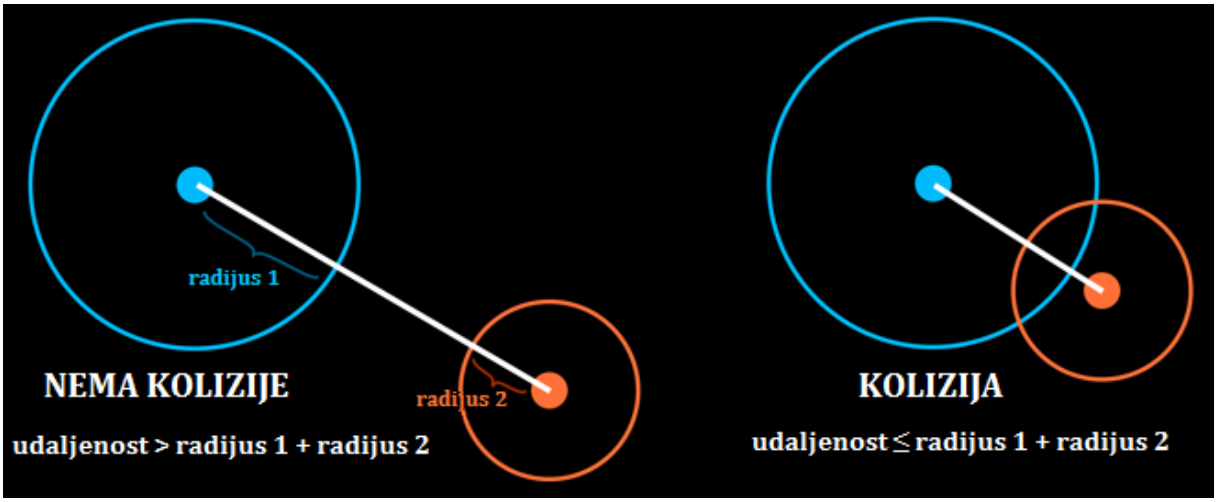
Destrukcija je oduvijek bila jedan od sastavnih dijelova računalnih igara – koncept nije ništa suvremeno, nalazimo ga već u najstarijim igrama kao što su „Super Mario“ (1985) i „Space Invaders“ (1987). Sagledano iz općeg konteksta, destrukcija je zapravo samo specifičan način interakcije te je oduvijek bila jedan od načina interakcije igrača s okolišem igre.

S druge strane, interakcija u stvarnom vremenu koncept je nešto novijeg datuma, a sa sobom je donio i problem destrukcije: kako odrediti početnu točku lomljenja, kako animirati svaki odlomljeni komadić, kako definirati novonastale odlomljene objekte u postojećem okolišu i sl. Jedna od prvih 3D igara u stvarnom vremenu koja je u potpunosti napravljena 3D alatima (koristi i 3D likove i 3D objekte te 3D okoliš) jest „Quake“[1].[2] Quake je nudio različite interakcije s raznim entitetima igre – objekte se moglo dodirnuti, a ako se u njih pucalo, učitavao se jednostavan skriptirani efekt lomljenja, uništavanja, odnosno destrukcije općenito. Slika 2.1. prikazuje ipak nešto noviji i napredniji primjer: scenu iz igre „Quake 4“ (2005) u kojoj se korištenjem odgovarajućeg oružja moglo rastopit zid i prijeći u drugu sobu.

[1] „Quake“ jest *first-person* pucačina koju je razvila tvrtka „id Software“ 1996. godine za Microsoft DOS.

Slika 2.1. *Scena iz igre „Quake 4“. [3]*

U igrama te generacije, interakcija se svodila uglavnom na skriptirane animacije ili potpuno uklanjanje objekta koji se našao kao prepreka. Destrukcija je bila moguća samo kod unaprijed određenih objekata te je razaranje svaki put izgledalo jednako, neovisno o mjestu udarca i sl. Fizika objekata ovdje se svodila na ništa više od jednostavne detekcije kolizije, uglavnom geometrijskog tipa, kao što je prikazano na slici 2.2. Prvi veliki pokušaj korištenja dinamičke destrukcije (eng. *dynamic destruction*) u 3D okolišu jest tehnologija GeoMod.



Slika 2.2. *Jedna od osnovnih metoda geometrijske detekcije kolizije. [4]*

2.1. GeoMod

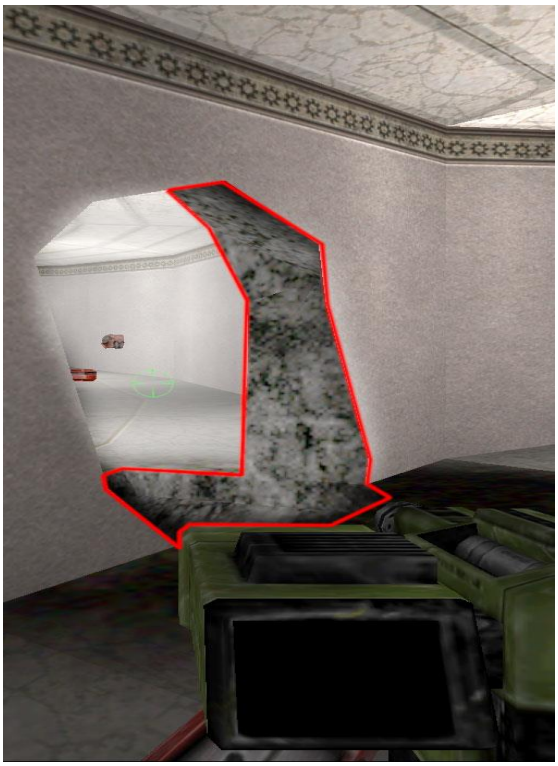
Geometrijska modifikacija (eng. *Geometry Modification*), odnosno skraćeno GeoMod, prvi je put upotrjebljen u igri „Red Faction“ (2001).[2] Tehnologija je dopuštala igraču da uništi velike dijelove okoliša do granice koja je prije bila nemoguća u stvarnom vremenu[3]. Umjesto da se, nakon detekcije kolizije, objekt obrisao i na njegovu mjestu učitao drugi, razorena verzija prvog objekta, GeoMod je stvorio novi prazan objekt koji je bio približne veličine i oblika uništenog područja. Radilo se zapravo o posebnom prozirnom objektu koji je skrivao objekt na kojem se učitava, ali dopušta vidljivost iza njega čime se simulirao prazan prostor nastao uništavanjem. Igraču je dana veća sloboda i mogućnost izbora, a igri je to donijelo veću popularnost. Usprkos tome, kritike su postojale. Fizika igre, i sve modifikacije općenito, ipak su bile vrlo ograničene i neusporedive su sa suvremenim grafičkim alatima. S obzirom da praznine nastale uništavanjem nisu zapravo bile praznine, već prazni objekti, teško se moglo simulirati djelovanje fizikalnih sila kao što je npr. gravitacija. To je dovelo do nekih potpuno nerealnih situacija, a jedna od njih prikazana je na slici 2.3.

Slika 2.3. *Scena iz igre „Red Faction“gdje su vatrenim oružjem uništeni dijelovi poda. [5]*

[2] „Red Faction“ razvila je tvrtka Volition Inc. Izdane su verzije za više platformi, prvo za konzolu PlayStation 2, a potom četiri mjeseca nakon i za PC.

[3] S dotad razvijenom tehnologijom, bilo bi potrebno ručno animirati svaki odlomljeni komadić te unaprijed definirati gotovo pa beskonačan niz kombinacija na koji igrač može uništiti pojedini objekt – iz kojeg kuta, kojim oružjem i sl.

Kako bi se prikrio loš efekt lomljenja, koristili su se efekti dima i prašine, npr. ako bi igrač uništio tanku liniju koja na slici 2.3. spaja dva dijela staze, platforma bi se srušila vertikalno prema dolje, uz dosta dima i prašine, nakon čega bi objekt bio zamijenjen nasumičnim manjim brojem statičnih krutih tijela, neovisno o masi prvotnog objekta. Od ostalih kritika valja još napomenuti kako je svaki objekt, bez obzira na materijal od kojeg je građen, uvijek imao isti uzorak materijala unutrašnjosti (prikazan na slici 2.4).



Slika 2.4. *Scena iz igre „Red Faction“, prikaz uzorka materijala unutrašnjosti. [5]*

U dorađenim verzijama GeoMod 2.0 i GeoMod 2.5 – predstavljene u igrama „Red Faction: Guerilla“ (2009) i „Red Faction: Armageddon“ (2011) – fizika igara dorađena je i izgledala je realističnije. Iako je u tom trenutku GeoModova budućnost izgledala kao budućnost čitave umjetnosti 3D destrukcije, njegov je razvoj prekinut zbog bankrota developera pod čijim je krovom funkcionirao Volition studio. [5] Iako se nadalo da će netko drugi nastaviti rad na ovoj tehnologiji, nikada nije objavljena tehnička dokumentacija o projektu pa se o tehničkoj implementaciji moglo samo nagađati. Vjerojatno je, ipak, da je GeoMod koristio boolove operatore za geometriju, pristup kojim se koristi CSG (Constructive Solid Geometry)[4] kojeg nalazimo i u suvremenim metodama.

[4] Više u ovom pristupu nalazi se u cjelini 3.1.1.

2.2. Frostbite Engine

U kontekstu efekta destrukcije, Frostbite enginei[5] posebno su bitna tehnologija. Ti su sustavi korišteni u svim izdanjima „Battlefield“ serijala igara, a njihov je razvoj počeo još 2007. godine s ciljem razvijanja potpuno dinamičke destrukcije u okolišu u kojem je svaki objekt lomljiv, uništiv. S obzirom da se radi o MMO (eng. *Massively Multiplayer Online*) računalnoj igri u stvarnom vremenu, realističnost nije bilo jedino na što valja paziti – pažnju je trebalo obratiti i na način na koji se nove tehnologije integriraju s renderom, posebice jer su ovi sustavi od samih početaka rađeni za razne platoforme: PC, Xbox 360 i PlayStation 3.

Prva igra koja je kreirana u Frostbite engineu jest „Battlefield: Bad Company“ (2008). Frostbite 1.0 koristio je posebni engine za destrukciju, Destruction 1.0, te je on bio zadužen za sve efekte lomljenja. [5] U igri je bilo moguće djelomično uništavanje građevina i uništavanje vozila, a uvedene su i neke eksperimentalne destrukcije: jednostavne deformacije terena, rušenje drveća, oštećivanje površina te razni mikroefekti kao što su čestice prljavštine.

S obzirom na uspjeh Frostbitea 1.0, razvoj se nastavio, uvedene su neke promjene i razvijen je Frostbite 1.5 u kojem su kreirane tri popularne igre: „Battlefield 1943“ (2009), „Medal of Honor“ MMO (2010) te „Battlefield: Bad Company 2“ (2010) čiji primjer prikazuje slika 2.5.

Slika 2.5. *Prikaz rušenja kuće u igri „Battlefield: Bad Company 2“. [6]*

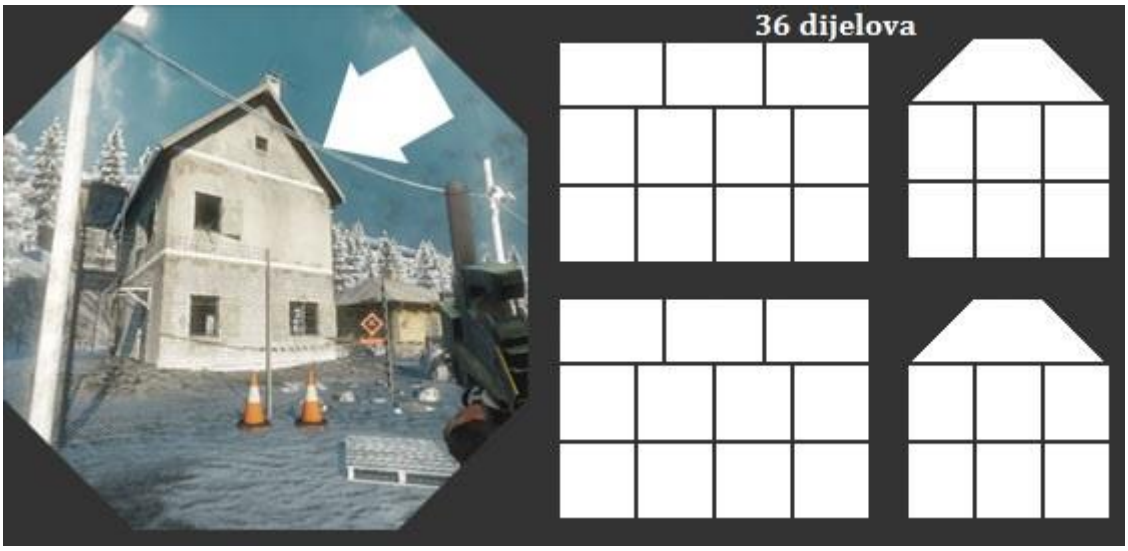
Frostbite 1.5 koristio je novi engine za destrukciju, Destruction 2.0 kojemu je glavna nova karakteristika mogućnost destrukcije čitavih građevina. Međutim, to je bilo moguće samo kod unaprijed određenih građevina koje su grafičari posebno modelirali i pripremili. [5]

[5] Prijevod engleskog termina tipa *game engine* sporno je područje. Budući da postoji nekoliko (jednako nezgrapnih, op.a.) prijevoda, a niti jedan nije opće prihvaćen, struka se koristi engleskom inačicom. U ovome radu postupa se jednako, a pod terminom *engine* podrazumijeva se sustav za izradu, razvoj i implementaciju u najširem smislu.

Na jednom igračem forumu, Den Kirson, jedan od developera igre „Battlefield: Bad Company 2“, sustav je objasnio ovako:

> „Građevina je skupina međusobno povezanih entiteta. Entiteti su hrpa zidnih i krovnih segmenata. Uništi dio, zid nestaje iza čestica dima i zamjenjuje ga rupa. Kad je [dovoljan] postotak segmenata uništen, građevina pokreće svoju animaciju urušavanja.“ [7]

Kao primjer navodi građevinu iza prve oznake MCOM u Port Valdezu i objašnjava kako je od ponuđenih 36 segmenata potrebno uništiti oko 26 kako bi se građevina urušila, dakle preko 70%. Kao što je vidljivo na slici 2.6, krov ne ulazi u postotak uništenih segmenata.



Slika 2.6. *Kirsonova slika kuće u Post Valdezu i podijele na segmente. [7]*

S obzirom da se Destruction 2.0 prezentirao kao napredak, u usporedbi sa, sada već dobro poznatom, CSG tehnikom korištenom u GeoModu 1.0, temeljenje destrukcije na skupini zasebnih entiteta čini se više kao tehničko i dinamičko nazadovanje. No, u ovom je slučaju bilo bitno kako rješenje funkcionira i izgleda unutar igre. Glavna prednost ovakve metode jest što se urušavanje inscenira na ograničen broj načina manipuliranjem manjim brojem segmenata – to destrukciju čini predvidljivom u svakom slučaju, a ishod u obliku unaprijed pripremljenih animacija igraču izgleda logično i realistično. Dokle god developeri imaju dovoljno dizajnera i grafičara koji se mogu nositi s ručnim konstruiranjem, modeliranjem i animacijom svake uništive građevine, igrači neće primijetiti razliku.

Slika 2.7. *Primjer dinamičke destrukcije u igri „Battlefield 3“. [6]*

Kao dinamički odgovor na 2.0 verziju, razvijen je željno očekivani Destruction 3.0 s kojim je kreirana igra „Battlefield 3“ (2011). Veliko zanimanje pobudila je njegova sasvim nova značajka nazvana mikrodestrukcija. Ova tehnologija ne samo da je omogućavala destrukciju većih dijelova i shodnu reakciju ako se uništi dovoljan postotak čitave građevine, već je igrač imao bitno precizniju ljestvicu oštećenja koja je mogao nanijeti i to strukturama bitno manjih dimenzija nego do sada. Svaki komad građevine koji je nastao nakon njezina uništenja ponaša se kao kruto tijelo i u interakciji je s igračem i ostatkom okoliša. Konačno, padajući komadi uništenih građevina i ruševina mogu ubiti igrača. Za razliku od „Bad Company“ serijala, u igri „Battlefiled 3“ ipak se nisu mogle uništiti sve strukture – stepeništa nekih betonskih zgrada i središta mnogih građevina odolijevaju potpunom uništenju, ali ne

zbog ograničenosti enginea, već zbog održavanja *gameplaya*[6]. Primjeri raznih uništavanja u igri „Battlefield 3“ nalaze se na slikama 2.7. i 2.8.

Slika 2.8. *Primjeri dinamičke destrukcije u igri „Battlefield 3“. [8][9]*

[6] Još jedan termin koji se, u nedostatku dostatnog prijevoda, koristiti u engleskoj inačici. Pod terminom *gameplay* podrazumijeva se radnja igre, način na koji je ona ispričana te način na koji se igra.

2.3. CryENGINE

CryENGINE jest Crytekov[7] sustav dostupan u četiri iteracije razvijane kroz jedanaest godina, a posljednja završna verzija enginea izdana je polovinom ove godine.

Prva verzija, CryENGINE 1, korištena je za izradu „FarCry“ (2004), igre koja je smatrana jednom od grafički najimpresivnijih igara svoga vremena. Svoj izgled duguje, međuostalim, tehnici koja je postala moguća zahvaljujući tada relativno novim alatima: *vertex-shaderu* 2.0 i *pixel-shaderu* 2.0. [5] Crytek tvrdi da je upravo ovaj engine prvi koji je koristio sjenčanje po pikselu[8] i HDR raspon kontrasta[9]. S obzirom da se engine pokazao uspješnim, a FarCry postao vrlo popularan, developeri su razvili nasljednika – CryENGINE 2 u kojem je kreirana igra „Crysis“ (2007). Ovaj je engine još jednom napravio prekretnicu na području grafike u stvarnom vremenu, fizike igre i lomljive aktive (eng. *Breakable Assets*). [5] Najimpresivnije značajke bile su one vezane uz fiziku mekih tijela, npr. lišća, te dinamički lomljiva stabla. Kao što je očekivano, dorađena iteracija CryENGINE 3 bila je u cijelosti još impresivnija te je u njoj kreirana igra „Crysis 2“ (2011) koja je osvojila i Njemačku nagradu za računalne igre u kategoriji Najbolja njemačka igra 2012. Sav rad i trud, te iskustvo spomenutih triju verzija, sabrano je u najnovijem engineu – CRYENGINE[10] čija je završna verzija izdana 19. lipnja 2015.

Primjeri scena iz igre „Crysis 2“ nalaze se na slikama 2.9. i 2.10, a primjeri iz „Crysis 3“ na slici 2.11. Obje igre kreirane su u CryENGINEu 3.

Slika 2.9. *Scene razorenog okoliša u igri „Crysis 2“. [10]*

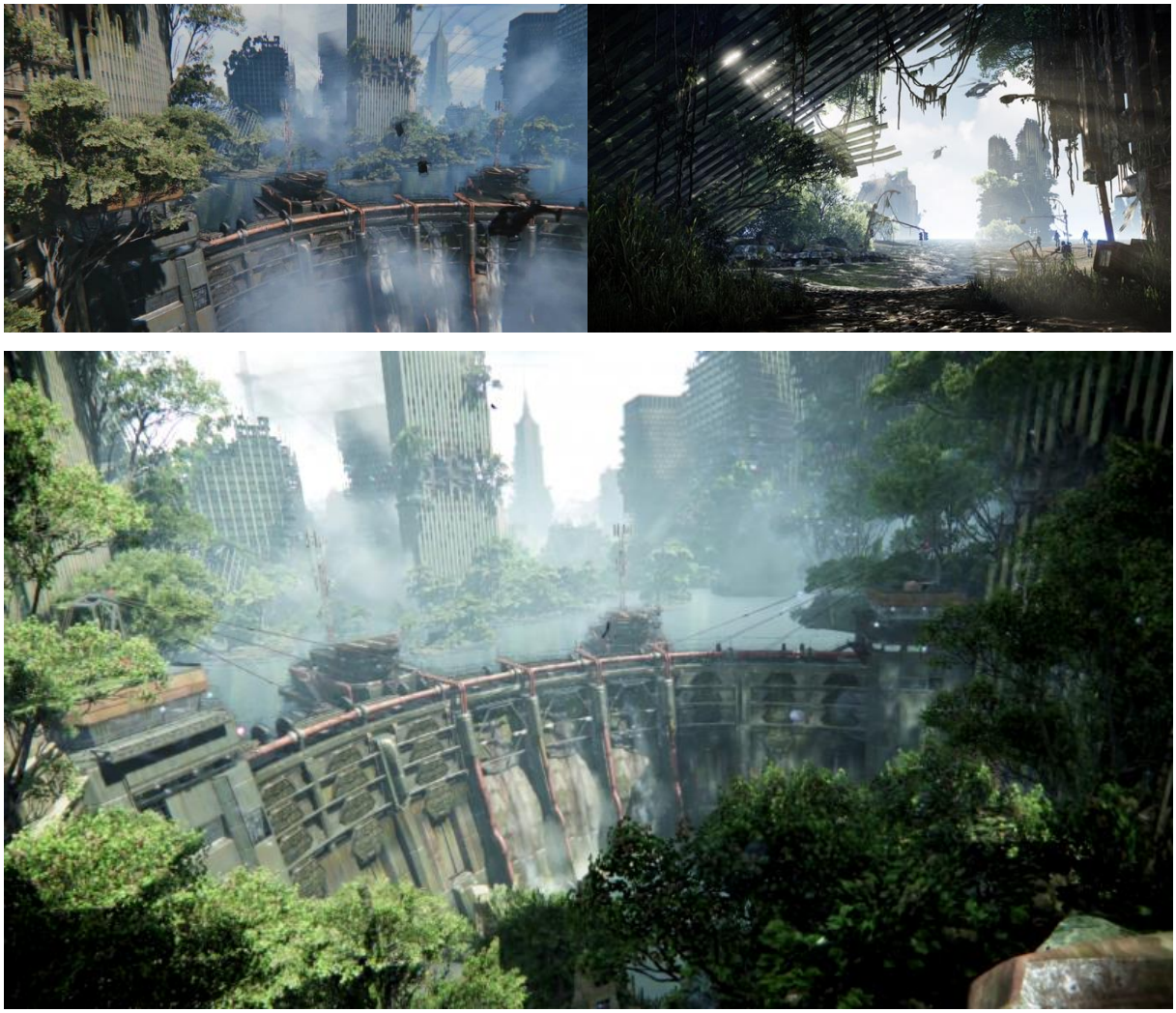
[7] Njemačka tvrtka Crytek najpoznatija je po serijalima first-person pucačina FarCry i Crysis.

[8] Sjenčanje po pikselu, eng. *per pixel shading*, podrazumijeva tehniku osvjetljavanja koja izračunava osvjetljenje za svaki pojedini piksel.

[9] HDR raspon kontrasta podrazumijeva tehniku luminacije koja izračunava kontrast koristeći veće raspone, npr. 32-bitna tekstura definirana decimalnim brojevima (eng. *floating point texture*) za pohranu mape osvjetljenja (eng. *lightmap*).

[10] Tvrtka Crytek odlučila je rebrendirati engine te je novu iteraciju izdala pod imenom „CRYENGINE“ bez broja. Ipak, radi praktičnosti praćenja nadograđenih verzija, razne numeričke oznake i dalje postoje. Posljednje, stabilno i završno izdanje, numerirano je s 3.8.1, dok je početna verzija bila 3.6.0.

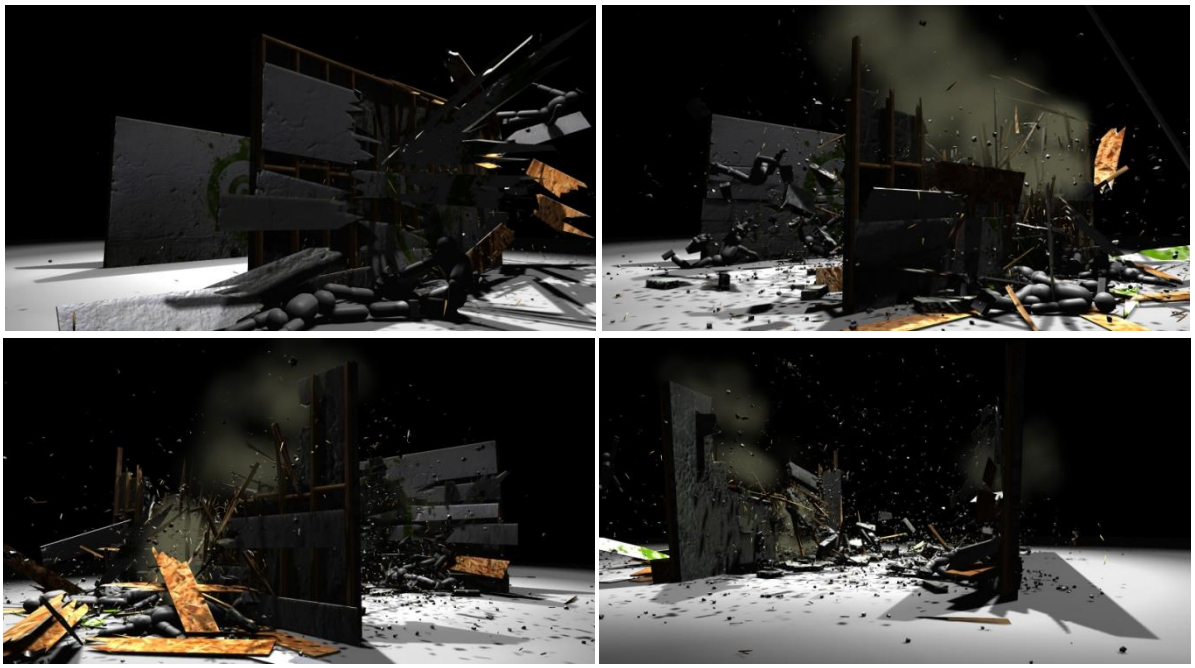
Slika 2.10. *Scena razorenog grada u igri „Crysis 2“. [10]*



Slika 2.11. *Scene razorenog grada u igri „Crysis 3“. [11][12]*

2.4. Unreal Engines

Idući bitan engine, ako ne i trenutno najkorišteniji od svih, jest UE3 (Unreal Engine 3) uz svoju besplatnu verziju UDK (Unreal Development Kit), čije su najnovije završne verzije izdane u veljači 2015. godine. U međuvremenu je izdana i završna verzija njihova nasljednika, UE4 (Unreal Engine 4), točnije 14. srpnja 2015. koja je također besplatna za korištenje. [13] Ovi enginei imaju zasebni modul za destrukciju koji direktno koristi NVIDIA APEX Destruction softver – vrhunski alat za lomljenje različitih predmeta, razbijanje zidova i sl., a koji neke od značajki nudi i u obliku plug-ina za Autodeskovu Mayu. Primjer razaranja korištenjem ovoga softvera nalazi se na slici 2.12.



Slika 2.12. *Uništavanje zida. Izvor slike: http://physxinfo.com [dostupno 9. rujna 2015.]*

Frostbite 2.0, CryEngine i Unreal Development Kit (UDK) danas su jedni od najnaprednijih sustava za 3D grafiku u stvarnom vremenu, pa onda i za destrukciju. [5] Korišteni su u brojnim igrama, na svim platformama, a svaki od njih problem destrukcije ima riješen na svoj način. Neki engini nude vlastite module za modeliranje i slične poslove, no u većini slučajeva, gotovi se elementi učitavaju u engine, oblikuje se okoliš, uređuju odnosi među entitetima i kreira čitav *gameplay*. Svemu tome prethodi dugotrajan rad u specijaliziranim alatima za modeliranje, teksturiranje, animiranje, zvuk i sl. pa onda i ishod rada u enginu ovisi također i o kvaliteti rada u povezanim alatima. S obzirom da iza različitih alata stoje različita teoretska rješenja kada je u pitanju teorija loma, po tom se ključu može dati njihov pregled.

3. Teorija loma

U prethodnom se poglavlju tema destrukcije promatra kroz lupu računalnih igara, a ovo poglavlje predmet sagledava kroz primjere sedme umjetnosti. Kod filmova je problem destrukcije nešto jednostavniji – zapravo se svodi na animaciju kojoj je cilj biti što realnijom, bez problema interakcije u stvarnom vremenu. Iako je iz jednadžbe maknut taj, inače otežavajući faktor, problem destrukcije u filmskoj industriji ipak nije banalan. Uništavanja ogromnih razmjera trebaju se doimati posve realnim i kvalitetno se integrirati sa stvarnim kadrovima. Većina sustava za simulaciju destrukcije temelji se na tehnologiji simulacije krutih tijela (RBS; eng. *Rigid Body Simulation*) koja se ponekad naziva i dinamikom krutih tijela (RBD; eng. *Rigid Body Dynamics*), ali i na novijem rješenju: analizi konačnih elemenata (FEA; eng. *Finite Element Analysis*)[11]. [14] Kada se govori o vizualnim specijalnim efektima (VFX; eng. *Visual Effects*) i računalno generiranim slikama (CGI; eng. *Computer Generated Image*), pojam destrukcije i kolizije uglavnom se svodi na pitanje dinamike krutih tijela. Bitno je pritom poznavati fiziku krutih tijela koja – za razliku od čestica, idealnih tijela bez mase i veličine, koje se mogu kretati samo u jednom od triju prostora i mogu se definirati vektorom – zauzimaju prostor i imaju geometrijska svojstva, kao što su središte mase, moment tromosti te, u ovom slučaju najrelevantnije, imaju šest stupnjeva pokretljivosti (translacija u smjeru svih triju osi te rotacija u trima smjerovima). Također, bitna karakteristika krutih tijela jest da se ne deformiraju, čime se razlikuju od mekih tijela. Ovakvim se temama, tehnologijama RBS-a i FEA-a bave neki od najvećih studija za VFX kao što su ILM, Imageworks, MPC, Framestore i Double Negative (Dneg). Posljednji od navedenih bio je i jedan od studija koji su radili na filmu iz 2009. godine čiji su efekti masovnih razaranja napravili prekretnicu u industriji – film „2012“. Najviše pažnje ipak je zaokupio rad studija Digital Domain. Scena uništavanja građevina razmjerom i opsegom izgledala je potpuno impresivno, a sve je napravljeno u nevjerojatnom vremenskom okviru s obzirom na obim posla. Prikazi odabranih kadrova te scene nalaze se na slikama 3.1. i 3.2.

[11] Radi se o tehnologiji koja je donedavno bila preskupa za sve osim za simulacije stvarnih inženjerskih problema, npr. sigurnosna ispitivanja u elektranama ili testiranje cjevovoda.

Slika 3.1. *Scene iz filma „2012“: razaranje Los Angelesa uzrokovano snažnim potresom, pogled iz zraka. [15]*

Slika 3.2. *Scene iz filma „2012“: razaranje Los Angelesa uzrokovano snažnim potresom, završni kadrovi potpunog uništenja. [15]*

Za slavnu scenu, Digital Domain studio kreirao je građevine koje nestaju u prašini, nebodere koji se urušavaju, ceste i autoputove koji pucaju te, naposljetku, ogromne komade kontinenta koji nestaju u oceanu. Ovakav zadatak prisilio je studio da problem promotri izvan granica procedura koje su do tada koristili – odlučili su se osloniti na rad alata „Bullet" – fizikalnog engina specijaliziranog za simulaciju detekcije kolizije te dinamiku krutih i mekih tijela – čime je njegova popularnost i prihvaćenost u industriji iznimno porasla. [15] Prema riječima Mohena Leoa, glavnog nadređenog za vizualne efekte u Digital Domainu:

> „Od početka smo znali da će simuliranje efekata biti najveći izazov i da ćemo trebati napraviti iznimno mnogo simulacija krutih tijela. Sagledali smo kakva rješenja trenutno imamo i nijedno od njih nije se moglo nositi sa zadatkom. Interno, nekoliko ljudi je bilo zainteresirano kreirati potpuno novi sustav baziran na Bulletu, projektu otvorenog koda. Bullet je vrlo jednostavan, ali je brz, stabilan i, zato što je otvorenog koda, može ga se dograđivati." [15]

Scena uništavanja[12] o kojoj se govori česta je referenca i u ozbiljnim teoretskim radovima. Njezina je uspješnost okrunjena šestim mjestom na popisu 25 najboljih računalno generiranih VFX-a svih vremena u „3D World" časopisu. Film „2012" u svakom je slučaju prekretnica za računalno generirane scene masovnih uništenja, veliki trijumf za Bullet, no njegova slava otprilike staje na tome – iako je mnogo zaradio, odnosno redatelju Ronaldu Emmerichu bio je komercijalno uspješan, nije doživio značajnije pohvale.

3.1. Vrste destrukcijskih procedura

Kao što je navedeno na početku poglavlja, postoji više tehnoloških rješenja za simulaciju destrukcije – destrukcijskih procedura (eng. *destruction pipeline*). Gledano u suštini, može ih se podijeliti u tri osnovne kategorije: osnovni RBS pristup, RBS nadopunjen specijaliziranim tzv. fizikalnim knjižnicama (eng. *physics libraries*) te FEA. Postoji i niz samostalnih komercijalnih rješenja, softvera i plug-inova namijenjenih za simulaciju destrukcije, među njima ističu se Houdini i 3ds Max, o kojima će biti nešto rečeno kasnije, te Maya, čije ćemo ugrađene opcije i plug-inove detaljnije opisati. Prije govora o samostalnim rješenjima, daje se sažeti prikaz destrukcijskih procedura općenito.

[12] Točnije, radi se o sceni potpunog uništenja Los Angelesa. Prvu polovicu scene napravio je studio Uncharted Territory, a drugu polovicu Digital Domain.

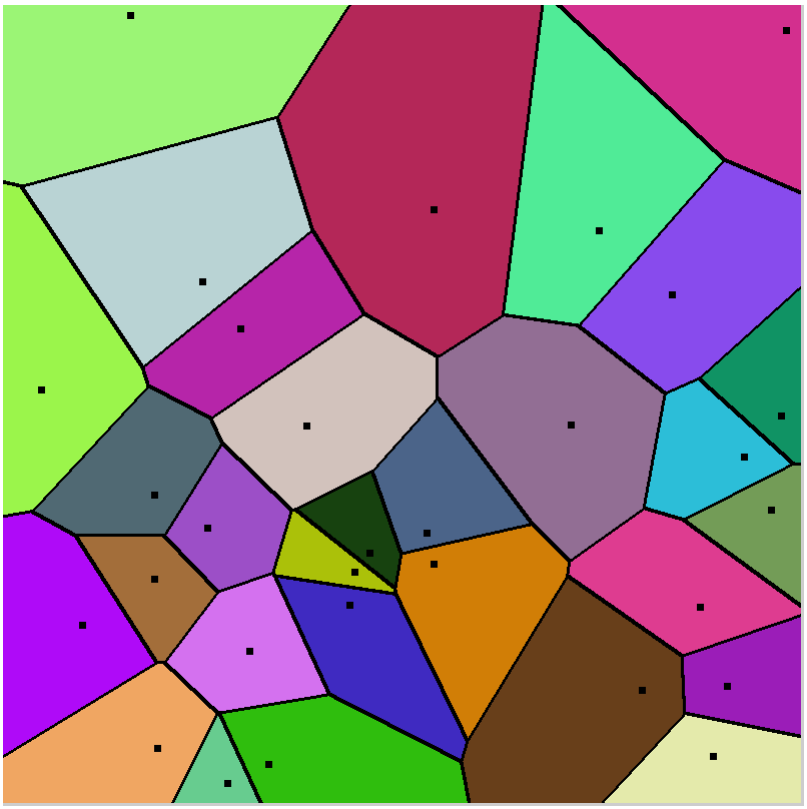
3.1.1. Osnovni RBS pristup

Ovaj se pristup sastoji od tri osnovna koraka: lomljenje objekata, definiranje ograničenja i dizajniranje koreografije te pokretanje simulacije i rješavanje detekcije kolizije. [14]

3.1.1.1. Lomljenje objekta – priprema geometrije

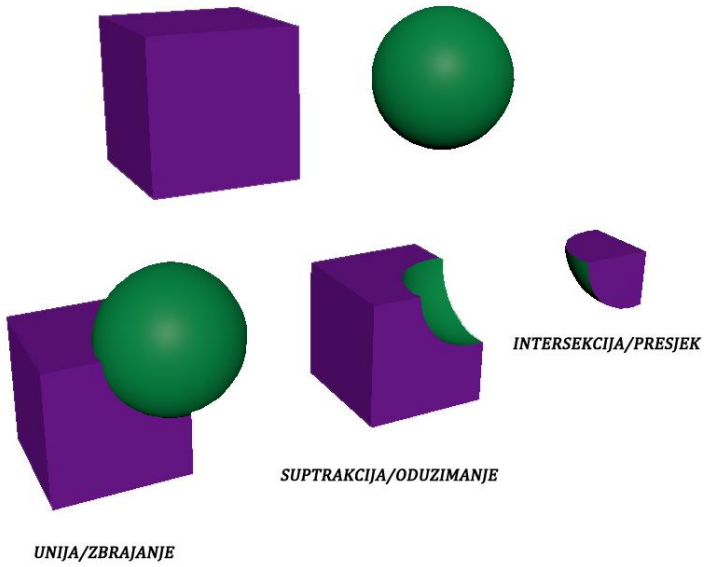
U prvom koraku, priprema se geometrija objekta. Erwin Coumans je na konferenciji SIGGRAPH 2011 u Vancoveru u Kanadi [16] iznio četiri metode kojima se to čini: voronoi dijagrami (1), CSG (2), konveksna dekompozicija (3) i tetraedarizacija (4).

Najčešće korištena jest upravo prva metoda. Voronoi dijagram matematički je koncept koji se koristi za kreiranje realističnih fragmenata. Princip je jednostavan, ali rezultat je vrlo realističan. Osnovan način konstrukcije jest da se između svake dvije točke A i B povuče simetrala dužine $\overline{AB}$. Primjer takvog dijagrama nalazi se na slici 3.3. Početne točke (ili čestice) generiraju se na mnogo načina, često i nasumice. Ovaj osnovni koncept je u srži mnogih RBS destrukcijskih procedura.

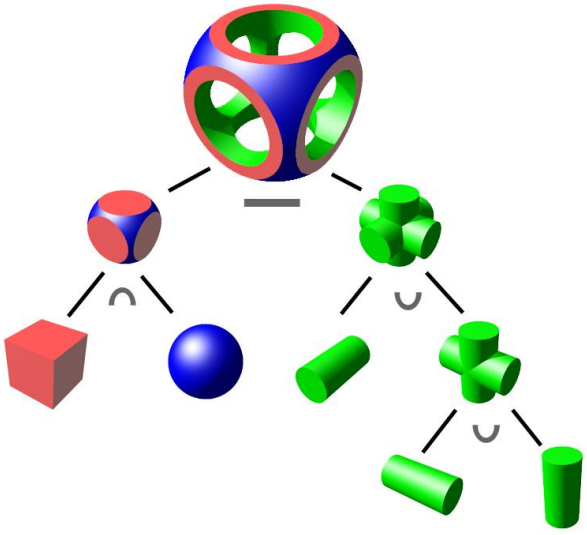


Slika 3.3. *Jednostavan plošni voronoi dijagram. [17]*

Druga tehnika jest CSG (eng. *Constructive Solid Geometry*) koja koristi boolove operatore [14], odnosno, kao što je pokazano na slici 3.4., osnovne operacije unije ili zbrajanja, suptrakcije ili oduzimanja i intersekcije ili presjeka s vrlo jednostavnim objektima. Tako se mogu obavljati volumne operacije između 3D modela te pomoću nekoliko jednostavnijih oblika ili objekata graditi kompleksne i realistične odlomljene oblike, kao što prikazuje primjer na slici 3.5.

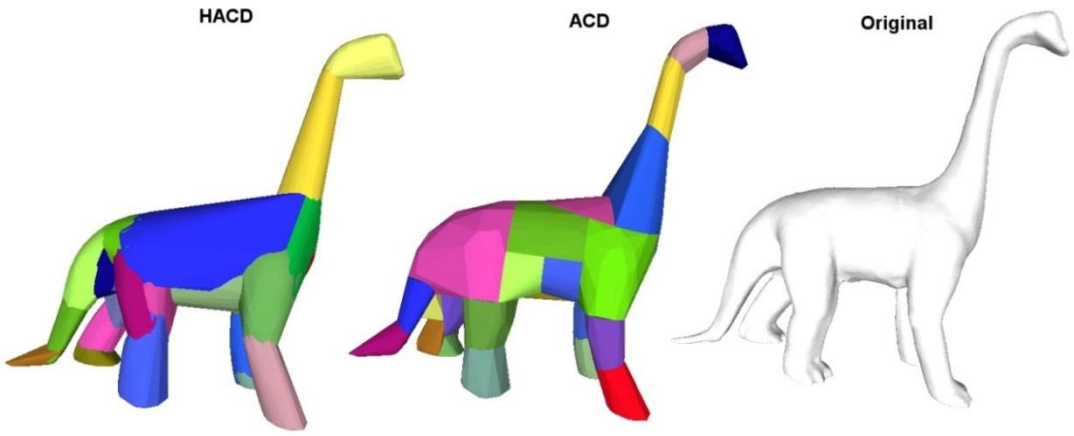


Slika 3.4. *Boolean operacije prikazane primtivima (kockom i kuglom).*



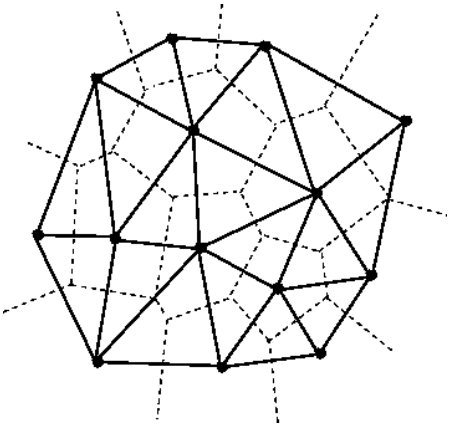
Slika 3.5. *Modeliranje kompleksnog objekta jednostavnim tijelima i boolovim operatorima. [18] (simboli: ∩ presjek, ∪ zbrajanje, – oduzimanje)*

Treća tehnika jest konveksna dekompozicija, a uglavnom se koristi za detekciju kolizije. [14] Ova tehnika može se provoditi ručno – grafičar dekomponira objekt ručno, koristeći jednostavne konveksne primitive kao što su, recimo kugle; ili automatski, korištenjem alata. Od automatskih tehnika, poznatije su približna konveksna dekompozicija (ACD; eng. *Approximate Convex Decomposition*) i hijerarhijska približna konveksna dekompozicija (HACD; eng. *Hierarchical Approximate Convex Decomposition*), a njihova je razlika vidljiva na slici 3.6.



Slika 3.6. *Prikaz tipova automatske konveksne dekompozicije. [19]*

Naposljetku, četvrta metoda jest tetraedarizacija, a često se koristi u FEA destrukcijskoj proceduri. Mreža se dekomponira u tetraedne elemente korištenjem Delaunayeve triangulacije[13] kao što je prikazano na slici 3.7. Postoji nekoliko implementacija otvorenog koda koje nude ovakvo rješenje, npr. „Netgen“ i „Tetgen“. [14] Jedan od destrukcijskih plug-inova za Mayu, Pixeluxov DMM (eng. *Digital Molecular Matter*), interno koristi Netgen kako bi napravio tetraedarizaciju.



Slika 3.7. *Delaunayeva triangulacija. [20]*

[13] Delaunayeva triangulacija jest jedinstvena triangulacija načinjena tako da niti jedna od definiranih točaka ne leži na opisanoj kružnici drugog trokuta.

3.1.1.2. Definiranje ograničenja i dizajniranje koreografije

Nakon što je geometrija pripremljena i izlomljena u unaprijed definirane fragmente, RBS zahtjeva da ih se međusobno poveže jer bi se u suprotnom razdvojili u bilo kojoj simulaciji s gravitacijom. Nekoliko je načina kako se to čini. Jedan od njih je da se definira veza između svakog pojedinog fragmenta sa susjednima. Ovako se dobiva maksimalna kontrola, ali može utjecati na performanse, posebice kod kompleksnih geometrija. Drugi način za definiranje veze između fragmenata jest korištenjem detekcije kolizije: definira se da kontakte točke između fragmenata postanu poveznice te im se postavi određeni prag tolerancije (eng. *treshold*).

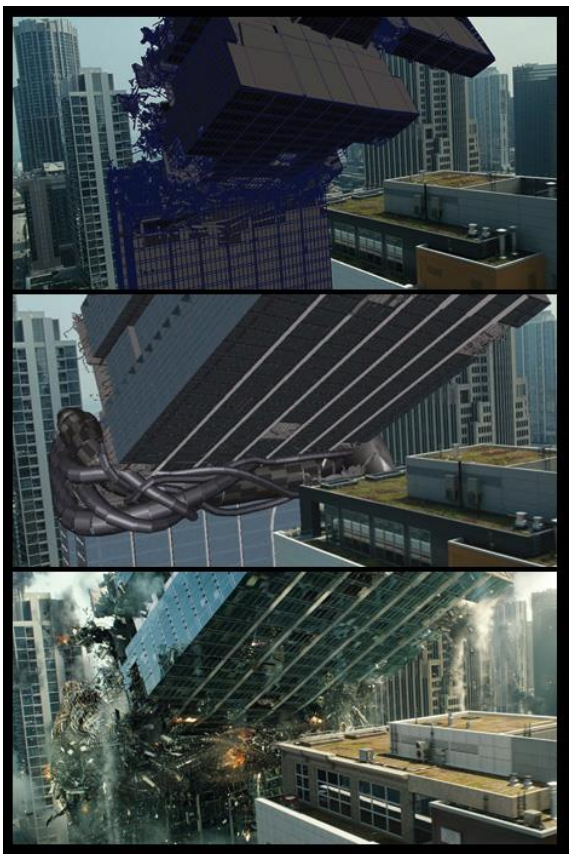
U ovom se koraku definira priroda fraktura te njihovo djelovanje. Može se odrediti njihovo kretanje, linije napuknuća koje ne pucaju do kraja i slični popratni efekti na materijalu. [14] Sve navedeno vrlo je delikatan posao i zahtjeva mnogo vremena, a nerijetko se dijelovi definiraju i programski (skriptom), kako bi se definirali svi mogući ishodi.

3.1.1.3. Pokretanje simulacije i rješavanje detekcije kolizije

Glavni problem ovog koraka jest kolizija, odnosno sudaranje. Jedan od najjednostavnijih način detekcije kolizije jest korištenje graničnog okvira (eng. *bounding box*). [5] Svaki objekt smješta se unutar jednostavnog graničnog okvira, a detekcija se bazira na principu prikazanom na slici 2.2. Iako je ovo najbrža i najjednostavnija metoda, njezini rezultati uglavnom nisu dovoljno dobri – kolizija se zbiva između graničnih okvira, a ne između objekata, što rezultira nerealističnim odbijanjem dvaju tijela jedno od druga, bez da je izgledalo da su uopće bili u kontaktu. Također, ova metoda možda je korisna za koliziju npr. odlomljenih kamenčića, ali nikako za koliziju stupa i automobila. Kompleksnije kolizije svaki alat rješava na svoj način, a on specifično ovisi i o objektima koji dolaze u kontakt. Ugrubo rečeno – dok se ponašanje i lom konveksnih objekata koliko toliko usklađeno rješavaju u većini alata (na dva ili tri glavna načina), konkavni objekti predstavljaju veći izazov. Neki engini koriste vlastite module za takve situacije, a nerijetko se one rade i ručnim animiranjem gdje je za ishod presudna gotovo isključivo vještina grafičara.

3.1.2. RBS i specijalizirane fizikalne knjižnice

Kao što joj i sam naziv govori, ova se destrukcijska procedura ne razlikuje mnogo od osnovnog RBS pristupa – zapravo se radi o RBS-u nadopunjenom, obogaćenom posebno dizajniranim algoritmima za različite aplikacije, odnosno za rješavanje specifičnih zadataka. Kada se govori o ovoj destrukcijskoj proceduri uglavnom se govori o knjižnici PhysBam razvijenoj na Stanfordskom sveučilištu u SAD-u i studiju za specijalne efekte Industrial Light & Magic (ILM). Njihov rad i razvijene metode dobile su brojne pohvale, kako na teoretskom polju, tako i u primjeni. [5] Ova je metoda korištena za razaranja u brojim poznatim filmovima: „Terminator 3: Pobuna strojeva“ (2003), „Zvjezdani ratovi: Treća epizoda – Osveta Sitha“ (2005), „Posejdon“ (2006) i „Svemogući Evan“ (2007). Kada se govori o ovoj destrukcijskoj proceduri i ILM-u, najbolje je spomenuti film „Transformeri: Tamna strana mjeseca“ (2011) u kojem je napravljena najdetaljnija i najkompleksnija scena uništavanja do tada [21], a čija je slava učvrstila ionako već dobro ukorijenjene temelje RBS modela. Kadar iz te scene prikazan je na slici 3.8., uz prikaz dvije faze u njegovoj izradi. Prikazuje se kako Driller ruši neboder. Usput rečeno, Driller je jedan od najkompleksnijih 3D modela ikada uvrštenih u filmove: ILM ga je izgradio od 30 milijuna poligona koji bi, kada bi ih se raširilo jednog do drugog, bili dugi 110 km. [22]



Slika 3.8. *Kaos u Chicagu: Driller ruši neboder. [22]*

3.1.3. Analiza konačnih elemenata (FEA)

Jednostavno rečeno, razlika između RBS-a i FEA-a je što se u RBS-u tijelo lomi na fragmente prije samog loma, a pomoću FEA frakture nastaju tek nakon loma te se pomoću niza povratnih informacija definira kako najbolje napraviti lomljenje. Gledano s fizikalne strane, FEA je ispravniji, točniji pristup destrukciji od RBS procedura. Analizom konačnih elementa i rješavanjem parcijalnih diferencijalnih jednadžbi određuje se dinamika elastičnih tijela, a deformacija i lom baziraju se na mehanici kontinuuma. 3D mreža aproksimira se pomoću zbira elemenata, uglavnom tetraedara, a izračun učinka sila i deformacija na kruta tijela uključuje i svojstva kao što su matrica krutosti, središte mase i sl. Procedura je razvijana 40-ih i 50-ih godina prošlog stoljeća, a u primjenu je ušla otprilike sredinom 70-ih. Tek odnedavno koristi se i za kreiranje VFX-a u filmovima i igrama. [14] The Moving Picture Company (MPC) najveći je studio koji koristi ovu metodu za izradu filmskih scena, a filmovi koji sadrže destrukcije napravljene ovom tehnologijom su npr. „X-men: Prva generacija“ (2011) i „Sucker Punch“ (2011) prikazan na slici 3.9.

Slika 3.9. *Destrukcija katedrale u filmu „Sucker Punch“. [38]*

Već spomenuti Pixeluxov plug-in, DMM, a o kojem se još govori i u cjelini 5.4.2, uz objašnjenu tetraedarizaciju koristi algoritam analize konačnih elemenata, naziva FEM sustav, a koristi se za lomljenje i deformaciju. Razvio ga je James F. O'Brien u sklopu svog doktorata na Kalifornijskom sveučilištu. FEA-ova primjena vrlo je realistična i kvalitetna, ali problem predstavlja cijena. Kao kompenzacija često nastupa DMM kako bi se smanjili troškovi, ali i povećala brzina rada, budući da je DMM ekonomičan i vrlo brz. [14]

4. Najpopularniji 3D alati

Gledano u suštini, svi samostalni komercijalni programi nalik su jedan drugome. Sadrže osnovno sučelje i osnovne alate, a doseg svoga rada uglavnom proširuju nepreglednim izborom skripta i plug-inova. Nekoliko je ključeva po kojima se ovi programi ipak razlikuju – od cijene, korisničke podrške i popularnosti do raspona osnovnih integriranih alata. Također, uvriježeno je mišljenje da su neki programi na određenom području bolji od drugih, i obratno. U skladu s time, daje se pregled srodnih 3D programa koji kotiraju najviše u konkurenciji u svim navedenim kategorijama, a s naglaskom na pitanje destrukcije. Ti su programi: Houdini, 3ds Max, Blender i Maya. U ovom poglavlju daje se sažeti uvod u prva tri, dok će se ostatak rada baviti posljednjim – Autodeskovom Mayom – kako teorijski, tako i praktično.

4.1. Houdini

Houdini, kojeg razvija tvrtka Side Effects Software, snažan je 3D alat koji se posebno ističe kada je u pitanju manipulacija krutim tijelima. Cijena godišnje licence ovisi o namjeni (obrazovna, osobna ili komercijalna) te o tipu paketa, a iznosi svugdje između 300 USD (obrazovna licenca 28. lipnja 2015.) i 10.000 USD (globalna komercijalna licenca 28. lipnja 2015). Houdini ima razvijen i vlastiti engine kojeg se kupuje odvojeno. [23]

Houdini je, kada je riječ o destrukciji, a i VFX-u općenito, doista dostojan biti na vrhu popisa komercijalnih 3D alata – vjerojatno je riječ o najpotpunijem i najstabilnijem softveru na tome području. Dolazi sa širokim spektrom integriranih alata koji su dostojni i za kompleksne zadatke lomljenja i simulacija eksplozija. Za slučajeve kada ugrađeni alati ipak nisu dovoljni, može se birati između niza samostalnih nadogradnji, odnosno plug-inova, od kojih se uglavnom svaki – plaća.

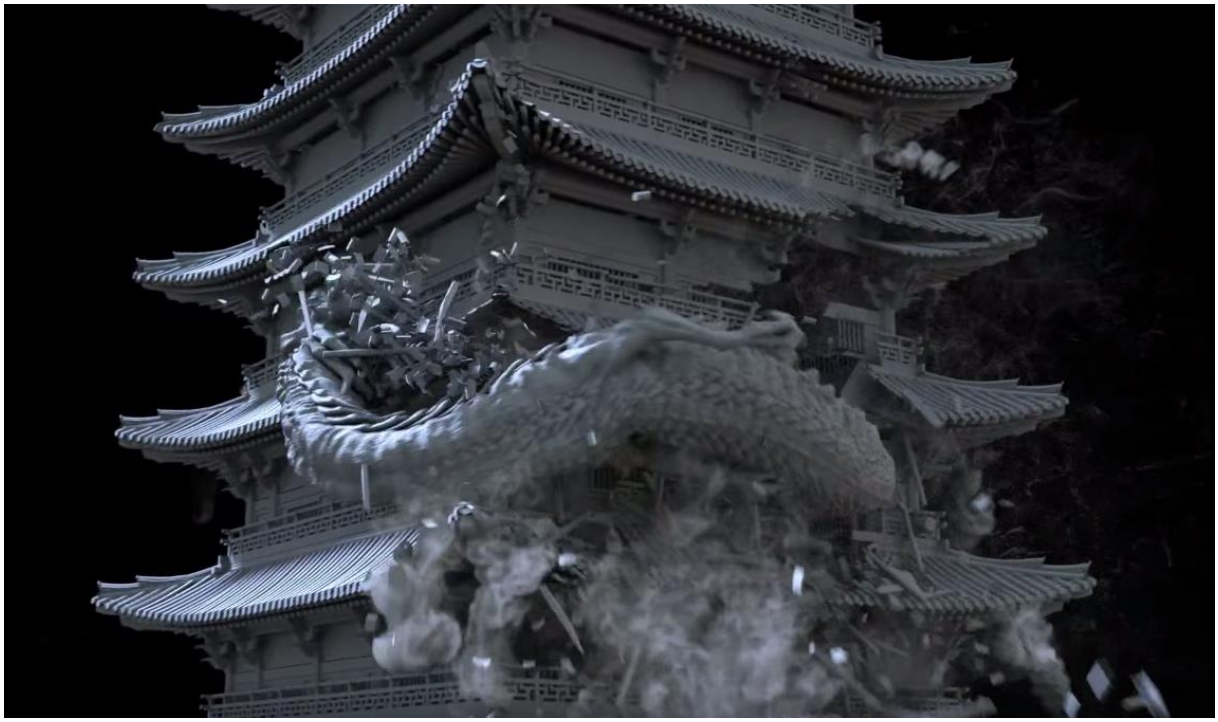
Softver pruža iznimnu kontrolu, ali i fleksibilnost i brzinu, što ga čini dobitnom kombinacijom. U zadnjih je deset godina doživio značajan porast popularnosti i postao sastavni softverski dio velikih VFX studija – Dneg je koristio Houdini pri radu na petom nastavku „Harry Pottera: Red Feniksa“ (2007) [24], A Pixar u „Priči o igračkama 3“ (2010). Zapravo, kako kaže Nicola Hoyle, Dneg Houdini koristi jednako koliko i Autodeskovu Mayu:

> „Kada je riječ o efektima, koristimo 50:50 Mayu i Houdini. U Mayi koristimo Dynamite vizualna rješenja i radimo simulacije tkanina. U Houdiniju vrtimo bilo što vezano uz zahtjevnija kruta tijela, a fluide, prašinu i vatru također radimo u Mayi.“ [14]

Osnovan način kreiranja destrukcije/loma u Houdiniju radi se kroz nekoliko koraka pri čemu je sav potreban alat integriran u softver. Prvo se animira kontrolni objekt kako bi se iscrtala linija loma. SopSolver snima animaciju i pohranjuje okvir po okvir kao kontrolnu geometriju na mreži. Deformacijom mreže stvara se izbočina koja prati lom. Tada softver prikuplja informacije vezano uz frakture. [25] Mreža dobiva dimenziju debljine i lomi se po principu voronoi dijagrama, a sve u skladu s informacijama koje su prikupljene u prethodnom koraku. Tako se dobiva unaprijed lomljena geometrija. Fragmenti se potom grupiraju na deformiranu mrežu, a pripisuje im se i atribut koji sadrži informaciju vezanu uz RBS i daljnju proceduru. Svaki fragment koji ima označeno da ga zahvaća dinamika (dakle, nije pasivan) lomi se, a fragmenti koji od njega nastanu grupiraju se. Potrebno je jedino definirati željene sile, dakle vektore kretanja te gravitaciju i lomljenje je gotovo. [14]

Slika 4.1. *Primjer kombiniranog loma krutih i mekih materijala. Screenshotovi napravljeni iz videa www.digitaltutors.com/tutorial/1091-Controlling-Your-Fractures-in-Houdini [dostupno 24. svibnja 2015.]*

Bitan korak za Houdini dogodio se u verziji 12 gdje je, osim raznih poboljšanja brzine rada, u njega integrirana i Bullet Physics nadogradnja. Houdinijev ugrađeni efekt loma i Bulletovo dinamičko lomljenje odlično se nadopunjuju i oboje u pravilu funkcioniraju na principu voronoi loma. Prednost ugrađenog alata za lom jest dobro podnošenje rada sa složenim oblicima, loše formiranom geometrijom, zahtjevnim preklapanjima i složenim početnim uvjetima. S druge strane, Bullet je idealan pri radu s pravilnom geometrijom i pruža jedinstvene prednosti pri rukovanju s ograničenim mrežama. [14] S obzirom na ovakvo nadopunjavanje, Bullet je Houdiniju bio prirodan odabir – no ne samo njemu – s obzirom da je program otvorenog koda, svoj je put pronašao i do svih drugih ovdje spomenutih 3D programa, uključujući Autodeskovu Mayu[14]. Glavni prigovor na Houdinijev ugrađeni lom uglavnom se svodi na teorijsku jednoličnost – značajniji korak izvan granica voronoi dijagrama još nikada nije učinjen.



Slika 4.2. *Efekt lomljenja i efekt dima u Houdiniju. Screenshot napravljen iz videa www.youtube.com/watch?v=zvUfdAyotRs [dostupno 9. rujna 2015.]*

[14] O Bulletu u Mayi više se govori u cjelini 5.4.1.

4.2. 3ds Max

Softver „3ds Max“, kao i Mayu, razvija „Autodesk Media and Entertainment“. Cijena godišnjih licenci jednaka je Mayinima, odnosno kreću se do 7.000 USD (4. kolovoza 2015), ovisno o paketu i namjeni. Iako je cjenovno podudaran Mayi, redovno se ističe njegova inferiornost, barem kada je u pitanju animacija. Uz Mayu, 3ds Max doima se kao Autodeskov *side project*, relativno nestabilan i prilično nepotpun, cijenu mu drži uglavnom Autodeskovo ime i globalna popularnost. Popularnost je zaradio uglavnom još sa starijim verzijama i jednostavnim te vrlo intuitivnim sučeljem i alatima. Zahvaljujući tome, cijene ga kao 3D softver ulaznog levela (eng. *entery level*) – prilično skupa inicijacija u svijet 3D grafike.

Priznat je kao snažan i potpun alat za modeliranje, ali slava mu otprilike tu i staje – nema razvijeno stabilno područje dinamike i ne nudi mnogo opcija vezanih uz animaciju. Sve navedeno rezultiralo je time da se brojni u 3D svijetu zapitaju koja je sudbina ovoga alata – planira li ga Autodesk prestati razvijati te se u potpunosti posvetiti Mayi? Na najvećim 3D grafičkim forumima mnogo se raspravljalo o tome, a Autodesk niti jednom nije dao službenu izjavu. [47] S obzirom da je prestanak rada na *Softimageu* Autodesk objavio tek u trenutku kada je izdao njegovu posljednju inačicu[15], jasno je da određen dio stručne zajednice očekuje kako bi se slična situacija mogla ponoviti. S jedne strane, dio stručnjaka očekuje Autodeskovu specijalizaciju na jednom 3D programu te vjeruju da će to biti Maya jer se na njezinim novijim iteracijama više radi te se njezino područje uporabe vidno širi, a s druge strane, ostatak stručnjaka vjeruje da 3ds Max drži čvrsto mjesto na području modeliranja i teško da će od tamo biti istisnut. Uz sve navedeno valja pridodati i kako se 3ds Max može koristiti isključivo na Windows OS-u što je relativno neobično za grafički softver, posebice uzevši u obzir da Autodesk ne radi na razvitku inačica za bilo koju drugu platformu, barem ne trenutno. [26] Za predmet ovoga rada ta opreka možda i nije presudna jer je gotovo pa sigurno da 3ds Max neće razvijati onaj modul rada koji se odnosi na destrukciju – više ili manje dinamičku. Konkretnije, stručnjaci uvjerenost u 3ds Maxov opstanak temelje na njegovoj specijalizaciji za područje modeliranja, možda čak i uže definirano na područje

[15] Autodesk je objavio kako će iteracija *Softimage 2015* biti ujedno i posljednja. Zadnja stabilna verzija izdana je 14. travnja 2014. godine, a korisničku podršku pružaju do isteka zadnje kupljenih programskih paketa, točnije do 30. travnja 2016. Svoj postupak objasnili su željom da se posvete manjem broju programa te tako kvalitetnije doprinesu grafičkoj i zabavljačkoj zajednici, a Softimageove korisnike pozvali su na migraciju na ostale Autodeskove programe.

arhitektonskog sadržaja. Bilo kakvo širenje na područje animacije ili dinamike općenito malo je vjerojatno. [47] Alate koje nudi za efekt lomljenja (ne)ima ugrađene već više od deset godina i njihova se uporaba za ovu svrhu nije mijenjala niti unaprjeđivala.

Kada je riječ o lomu, 3ds Max zapravo nema ugrađenog alata već se lom rješava raznim plug-inovima: uglavnom oni koji postoje za Mayu, postoje i za 3ds Max tako da je izbor zaista velik i raznolik, a neki od boljih ovdje su i opisani. S obzirom da se plug-inovi plaćaju, a cijena im varira svugdje između 5 USD i 800 USD, ako je lom nužno napraviti s alatima koje softver nudi, to se može napraviti kombinacijom boolovih operatora i alata za rezanje. [26]

Slika 4.3. *Primjer loma u 3ds Maxu napravljenog alatom za rezanje. [26]*

Alat se naziva *ProCutter Compound Objects* i dostatan je čak i za zahtjevnija lomljenja, ali, kao što je vidljivo na slici 4.3, zahtjeva mnogo ručnog rada i poznavanje načina na koji se zamišljeni materijal lomi. *ProCutter* se može koristiti za simuliranje eksplozija, raspadanja, lomljenja, montaže, fragmentacije ili za kreiranje puzzle efekta. [26]

4.3. Blender

Neizbježna točka na ovome popisu u svakom je slučaju Blender – 3D grafički program otvorenog koda čija je inicijalna verzija izašla još 1995. godine. Činjenica da je besplatan čini ovaj softver vrlo primamljivom alternativom u svijetu 3D računalne grafike, posebice uzevši u obzir da i vrlo primjenjiv i snažan alat. Dolazi s ugrađenim Bullet physics dodatkom (eng. *add-on*), ali i vlastitim ugrađenim alatom namijenjenim za lomljenje.

Slika 4.4. *Primjer voronoi loma u Blenderu. [27]*

Primjer na slici 4.4. napravljen je Blenderovim alatom za efekt lomljenja koji se naziva *Cell Fracture* i u početnim je postavkama isključen, a aktivira ga se u kategoriji *Add Ons*. [28] Bez njegove aktivacije objekti se mogu lomiti još jednostavnije – naredbom *divide*, ali tada se radi o lomu gotovo bez ikakve kontrole. Kao i Bullet, *Cell Fracture* lomi objekte po principu voronoi dijagrama.

Lom se, bez obzira na tip loma objekta, raznim tehnikama može finije podesiti, recimo korištenjem čestica (eng. *particles*) ili, još intuitivnije, pomoću olovke masnog traga (eng. *grease pencil*). [29]

5. Autodesk Maya

Autodesk Maya (san. माया [*māyā*] – iluzija) jest 3D grafički softver koji funkcionira na više operacijskih sustava: Windows, Mac OS X i Linux. Prva inačica, Maya 1.0, izdana je 1998. godine, a Autodesk je projekt kupio 2005. godine i još ga uvijek razvija. Najrecentnija iteracija, Maya 2016, izašla je 15. travnja 2015., a s njome i brojne nadogradnje i proširenja. [30] Već u prvom tjednu mišljenja o novoj verziji uglavnom su pozitivna: prepoznato je unaprjeđenje sustava i brzine rada te dorađeno sučelje, a kritike su išle na račun nekompatibilnost s nekim karakteristikama modeliranja i teksturama u starijim verzijama. Od interesa za predmet ovog rada, osnovni modul za simulaciju fizikalnih zakonitosti nije se mijenjao, već su samo neke značajke dodane, a neke dorađene i pregrupirane. Posebno se radilo na području Bifrosta te XGena. [30] Očito je, također, da se dodatno radilo na općoj integraciji Bulletovog fizikalnog engina čime ga se polako promiče na čelo Mayinih planova kada je riječ o simulaciji dinamike krutih tijela. [31] O Bulletu u Mayi više se informacija daje kasnije u radu, nakon pobližeg prikaza odabranih dijelova Mayinog dinamičkog modula.

5.1. Dinamička polja

U Autodesk Mayu ugrađena je podrška za rad s poljima koja simuliraju fiziku stvarnih pojava i sila – dinamička polja (eng. *dynamics fields*). Polja su područja sile koje koristimo za animiranje (i simuliranje) gibanja čestica, nParticlesa, nClotha, mekih tijela, krutih tijela, fluida i vlasi. Meko tijelo jest poligonalna površina, NURBS površina, NURBS krivulja ili rešetka (eng. *lattice*). Kruto tijelo jest poligonalna ili NURBS površina. Polje se može kreirati na jednom od takvih objekata ili kao samostalna sila. [32]

Svako dinamičko polje određene je klase i određene vrste. Kao što pokazuje tablica 5.1, razlikujemo tri osnovne klase dinamičkih polja te devet različitih vrsta. Klase i vrste kombiniraju se proizvoljno, ovisno o željenom učinku, a ograničenja postoje ovisno o prirodi objekta na koji sila utječe. Uz sve spomenuto, različite se vrste mogu koristiti zajedno kako bi se dobio što realniji rezultat.

D I N A M I Č K A P O L J A

KLASE	VRSTE
1. samostalno polje	**a)** zračno polje
2. objektno polje	**b)** vlačno polje
3. volumno polje	**c)** gravitacijsko polje
	d) Newtonovo polje
	e) radijalno polje
	f) polje turbulencije
	g) uniformno polje
	h) vrtložno polje
	i) volumno polje

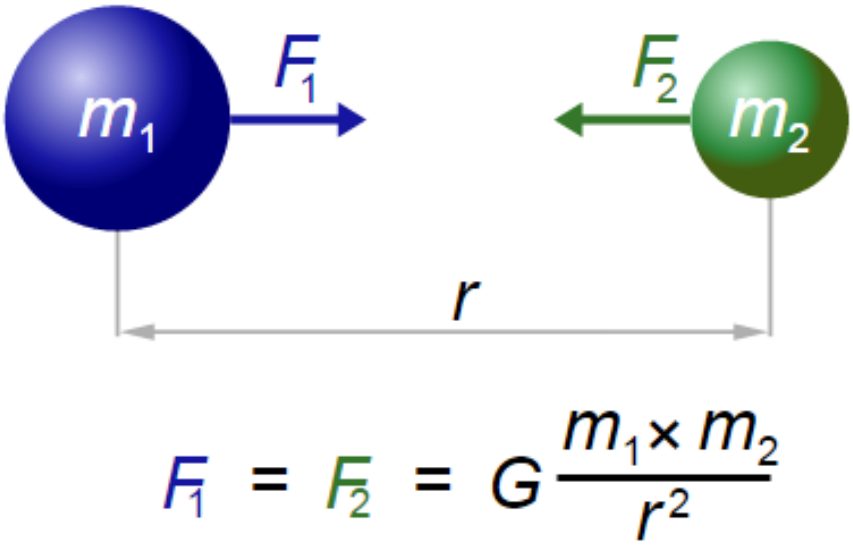
Tablica 5.1. *Podjela Mayinih dinamičkih polja na klase i vrste.*

5.1.1. Klase

Samostalna polja (1) utječu na objekte iz fiksne ili pokretne pozicije na sceni te nisu u vlasništvu geometrije. S druge strane, objektna polja (2) u vlasništvu su objekta na koji i utječu. Taj objekt može biti poligon, NURBS krivulja, NURBS površina, čestica, rešetka ili krivulja na površini. Kod čestica i krivulja, polje se može definirati za cjelinu ili za samo jedan dio: polje zahvaća sve čestice, ili samo pojedine; polje zahvaća čitavu krivulju, ili samo njezine odabrane dijelove. Naposljetku, sile mogu biti definirane u obliku volumnog polja (3), a njegovi mogući oblici su kugla, kocka, valjak, stožac ili prsten. [32]

5.1.2. Vrste

Zračno polje (a) simulira razne utjecaje kretanja zraka. Kada polje zahvaća objekt, on usporava ili ubrzava kako bi svoju brzinu uskladio s brzinom polja. Vlačno polje (b) djeluje poput protusile izazvane trenjem ili kočenjem. Gravitacijsko polje (c), kao što je očito iz njegova imena, simulira Zemljinu gravitacijsku silu, odnosno ubrzava objekt u zadanom smjeru. Newtonovo polje (d) objekte privlači prema sebi te se uglavnom koristi za simulaciju centripetalne sile, ali i za različite simulacije sudara. Ime je dobilo prema Newtonovom zakonu gravitacije koji tumači kako se bilo koja dva tijela privlače uzajamnom silom, proporcionalnom umnošku njihovih masa, a obrnuto proporcionalno kvadratu njihove međusobne udaljenosti. Dakle, što su tijela veća, sila je snažnija, te što su tijela udaljenija, sila je slabija. Grafički prikaz ovoga zakona prikazan je na slici 5.1.



Slika 5.1. *Grafički prikaz Newtonovog zakona gravitacije.*
Izvor slike: https://hr.wikipedia.org/wiki/Newtonov_zakon_gravitacije
[dostupno: 1. lipnja 2015.]

Radijalno polje (e) gura objekte ili ih vuče prema sebi, a snaga ovisi o udaljenosti od polja. Polje turbulencije (f) uzrokuje nepravilnosti u kretanju objekata koje zahvaća. Često se kombinira s drugim poljima te se mnogo koristi za oponašanje gibanja fluida. Uniformno polje (g) gura objekte u jednom, unaprijed određenom smjeru. Vrtložno polje (h) vuče objekte u kružnom ili spiralnom smjeru. Te konačno, volumno polje (i) omogućuje pomicanje predmeta u različitim smjerovima i unaprijed definiranom volumenu. Tipično se koristi za eksplozije, solarne baklje, razne efekte koji uključuju kretanje preko ili oko prepreka i sl. [32]

Ovaj sažeti prikaz klasa i vrsta dinamičkih polja doista je samo uvod u mnoštvo opcija koje Mayin dinamički modul nudi. Kombiniranjem raznih značajki i finim podešavanjem karakteristika, može se postići željeno simuliranje stvarnih fizikalnih pojava, ili pak maštovito kreiranje nekih nadrealnih – sve što bi bilo zahtjevnije tradicionalnim animiranjem.

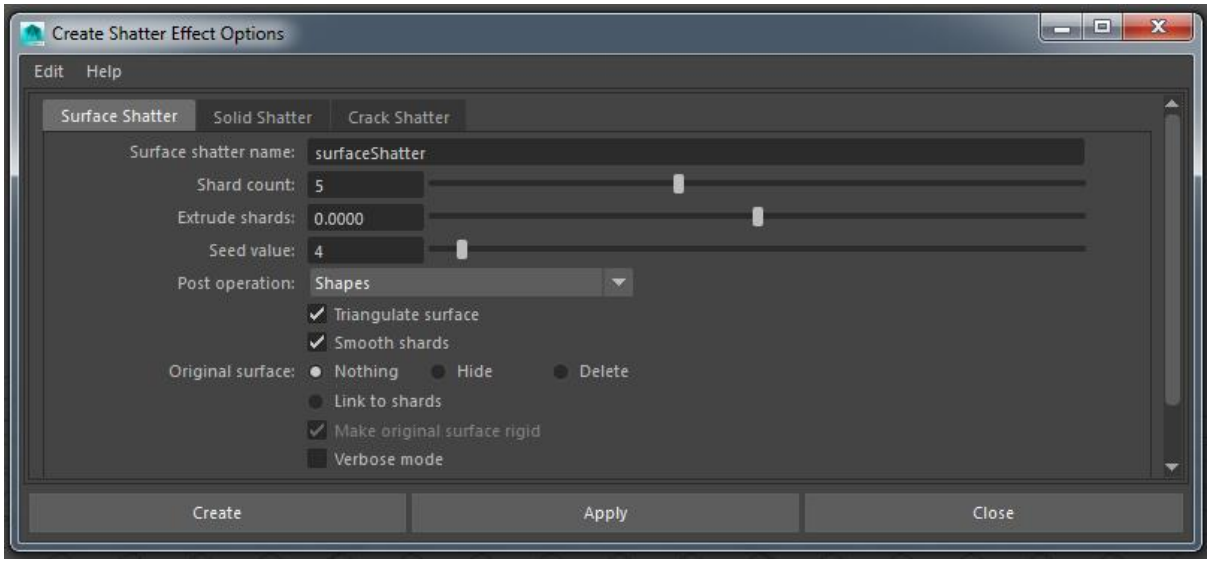
5.2. Efekt lomljenja

U područje dinamičkog modula pripada i simulacija destrukcije, a Autodesk Maya dolazi s ugrađenim alatom za efekt lomljenja (eng. *shatter effect*) – alat kojemu je, u odnosu na tradicionalni ručni način animiranja, cilj olakšati simuliranje lomljenja, napuknuća, razbijanja, generalno destrukcije.

Princip rada ovoga efekta u osnovi je vrlo jednostavan – pri kreiranju ovog efekta, objekt se duplicira i lomi na više fragmenata. [32] Ovisno o odabranom tipu loma, nad stvorenim fragmentima mogu se kreirati dinamičke sile ili se njihovo kretanje može animirati ključnim okvirima (eng. *keyframe*). Efekt pruža tri tipa loma: lom šupljeg tijela, odnosno lom površine (eng. *surface shatter*), lom punog tijela (eng. *solid shatter*) i napuknuće (eng. *crack shatter*).

5.2.1. Lom površine

Lom šupljeg tijela, odnosno lom površine efekt je koji odabrani objekt lomi prateći njegove poligonalne rubove. Površinskim lomljenjem objekta nastaju fragmenti koji su i sami individualni poligoni. Ovaj se efekt može primijeniti na NURBS ili poligonalno tijelo, a površine mogu biti i otvorene i zatvorene. Ako se efekt primjenjuje na NURBS površinu, objekt će prije loma biti opločen (eng. *tessellation, tiling*), a fragmenti nastali lomom bit će poligoni. Kao što je vidljivo na slici 5.2, postoji nekoliko postavki kojima se može upravljati rezultatom površinskog loma.



Slika 5.2. *Prozor za kreiranje loma površine u Mayi.*

U polju *shard count* definira se željeni broj fragmenata. Ako je upisani broj veći od broja poligona objekta koji se lomi, Maya stvara fragment od svakog pojedinog poligona kako bi se postigla željena količina. U slučaju da je količina fragmenata vrlo velik broj, stvaran broj nastalih fragmenata ne mora biti točan, već približan. Polje *extrude shards* prima pozitivne i negativne decimalne vrijednosti, a određuje debljinu površinskih fragmenata. Pozitivne vrijednosti odguruju površinu prema van kako bi se povećala debljina, a negativne vrijednosti privlače površinu prema unutra. Pomoću polja *seed value* može se odrediti vrijednost nasumičnom brojevnom generatoru, odnosno, ako je tu upisana vrijednost nula, onda će svaki put rezultat lomljenja biti drukčiji. Za upisane vrijednosti veće od nule, rezultati lomljenja svaki će put biti jednaki. Kako bi se definiralo što će fragmenti biti, koristi se polje *post operation*. Ono u padajućem izborniku nudi pet opcija: *shapes, rigid bodies with collisions off, soft bodies with goals, soft bodies with lattice and goals* i *sets*.

Odabirom *shapes*, odnosno oblika, nastaju poligonalni fragmenti na koje se može primijeniti bilo kakva animacija, a idealno je za animiranje ključnim okvirima. Kada se na fragmente žele primijeniti neke dinamičke sile, dobro je pri kreiranju loma odabrati jednu od idućih triju opcija. No, ako je oblicima ipak potrebno dodati dinamička polja, potrebno je uzeti u obzir da ih u tome slučaju Maya automatski pretvara u aktivna kruta tijela s uključenom detekcijom kolizije. S obzirom da su fragmentirani oblici međusobno u kontaktu, uključena detekcija kolizije može dovesti do nepoželjnih rezultata i pogrešaka pri djelovanju dinamičkih sila. Kako bi se izbjegla ovakva situacija, dobro je ručno isključiti detekciju kolizije.

Odabirom opcije *rigid bodies with collisions off* fragmenti će biti kruta tijela s isključenom kolizijom. Kolizija se isključuje kako bi se izbjegle pogreške koje mogu nastati kada se fragmenti dotiču. Takva je postavka idealna ako se na fragmente primjenjuje neka dinamička sila koja ih pomiče, a kolizija se onda uključuje kada fragmenti više nisu u kontaktu, odnosno kada su razmaknuti.

Postoje dvije opcije u kojima su fragmenti meka tijela. Prva jest *soft bodies with goals* koja kreira fragmente: meka tijela koja se deformiraju kada se na njih primjene dinamičke sile. Opcija ima još jednu karakteristiku – dodaje i tzv. ciljne objekte (eng. *goal objects*) koji su kopije fragmenata i služe kako bi se kontroliralo i preciznije odredilo deformiranje. U istu svrhu, kreiraju se i opruge (eng. *springs*) koje mekim tijelima osiguravaju unutarnju strukturu.

Druga takva opcija jest *soft bodies with lattice and goals* koja lomi objekt na fragmente, svakom fragmentu dodjeljuje rešetku i pretvara je u meko tijelo. Kada je rešetka meko tijelo, na nju mogu utjecati dinamička polja, a tada ona deformira fragmente u skladu s

utjecajem polja. Softver također dodaje i ciljne objekte i opruge, s istom svrhom kao i u prethodnom modelu. Deformacija pomoću rešetke kudikamo je brža, posebice kada se radi na objektu s kompleksnom geometrijom. Također, rešetki je jednostavnije dodati oprugu, čime se ostvaruje ravnomjernija treptavost objekta. No, valja prigovoriti da, ako se planira simulirati koliziju, rezultati mogu biti nerealistični s obzirom da se kolizija zapravo događa između rešetki, a ne između fragmenata.

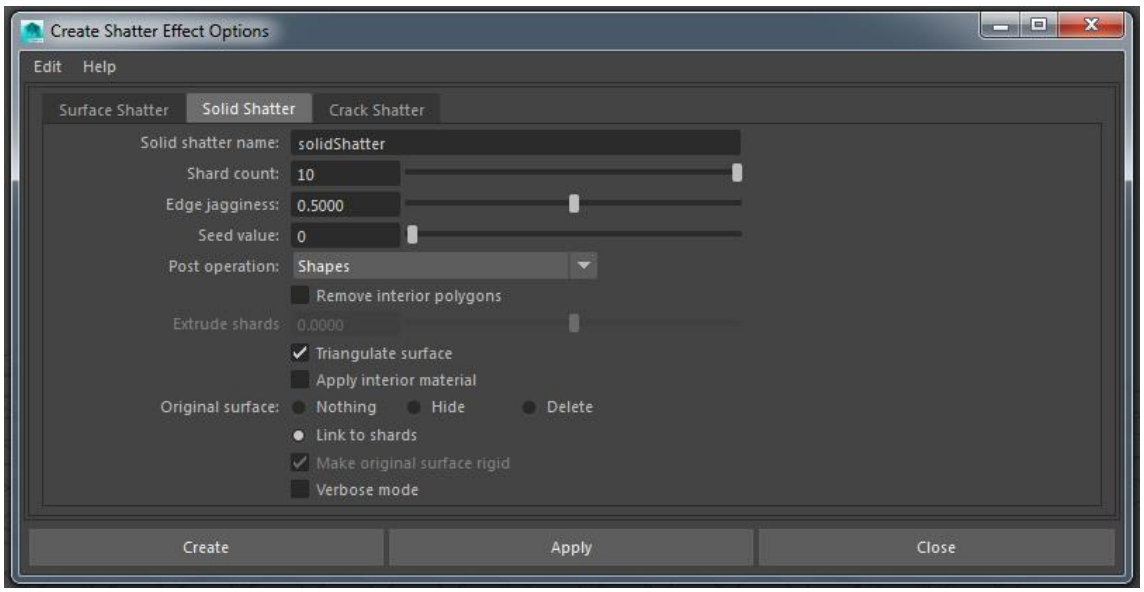
Posljednja opcija koju se može izabrati u padajućem izborniku polja *post operation* jest *sets*, odnosno skupovi. Odabirom ove opcije, objekt se zapravo ne lomi, već samo grupira poligone fragmenata tako da prvotni objekt ostaje nerazdvojen. Ovakvi se skupovi koriste kao i svi drugi skupovi u Mayi – može im se dodijeliti različita boja i sl. [32]

Izbornik za kreiranje efekta lomljenja nudi i triangulaciju čitave površine (eng. *triangulate surface*) – ovisno o kompleksnosti modela, odaziv može biti relativno dug. Odabirom opcije *smooth shards* redistribuiraju se poligoni na rubovima fragmenata čime oni postaju glatkiji. Bitno je napomenuti kako ovaj odabir rezultira i dužim vremenom potrebnim za obradu. To možda nije bitno kada se radi na jednostavnim objektima, ali kod kompleksnih geometrija, vrijeme odaziva može biti i trostruko duže, čime se znatno opterećuje rad računala.[16] Opcija *original surface* služi kako bi se nakon kreiranja loma upravljalo izvornim objektom, a na odabir su ponuđene tri opcije: izvorni se objekt ostavlja na sceni; izvorni se objekt skriva, ali još uvijek postoji na sceni; ili se pak izvorni objekt briše. Moguće je odabrati i opciju *link to shards* koja stvara nekoliko poveznica između izvornog objekta i fragmenata. Time se omogućuje kontrola vidljivosti izvornog objekta i fragmenta pomoću jednog atributa na transformacijskom čvoru izvornog objekta. Opcija ima različite postupke i rezultate ovisno o tome primjenjuje li se na kruta ili meka tijela. [32] Opcija *make original surface rigid* aktivira se samo kada se u polju *post operation* odabere model koji kreira kruta tijela i kada se odabere opcija *link to shards* – u ovim okolnostima može se odabrati da izvorni objekt bude aktivno kruto tijelo. Naposljetku, može se odabrati opcija *verbose mode* koja bilježi detaljniji opis postupka u prozoru s naredbama i povratnim informacijama.

[16] O vremenu potrebnom za odaziv i općenitom utjecaju efekta loma na računalne performanse više se daje u cjelini 5.5.

5.2.2. Lom punog tijela

Lom punog tijela lomi površinu objekta, ali zadržava i unutarnju strukturu te stvara pune fragmente. Lomljenje se ne stvara uzduž poligonalnih rubova što rubove fragmenata čini realističnijima. Ovaj se efekt može primijeniti na NURBS i poligonalne objekte, a površine moraju biti nužno zatvorene. Kao što je vidljivo na slici 5.3, većina postavki kojima se upravlja rezultatom loma punog tijela jednaka je onima za lom površine.

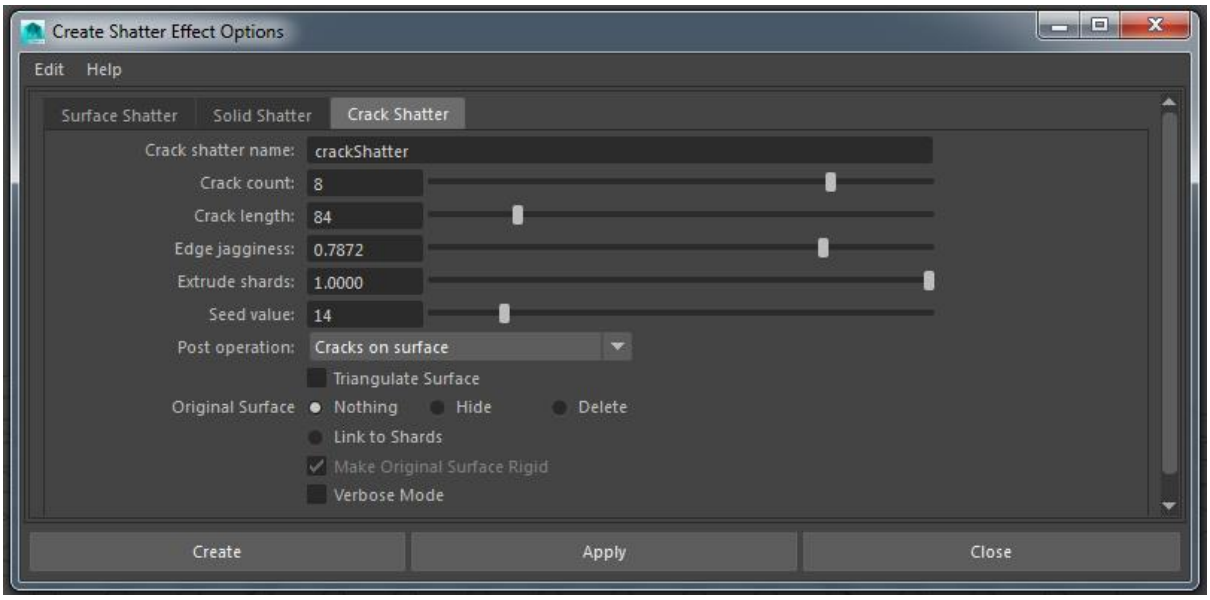


Slika 5.3. *Prozor za kreiranje loma punog tijela u Mayi.*

Polja jednaka onima koja su opisana kod loma površine jesu *shard count*, *seed value*, *extrude shards*, *triangulate surface*, *original surface*, *link to shards*, *make original surface rigid* i *verbose mode*. Razlike postoje u polju *post operation* koje nudi sve modele kao i kod loma površine, osim skupova (eng. *sets*), a nude se i tri posve nova polja. U prvom od njih, *edge jagginess*, može se odrediti neujednačenost odlomljenih rubova. Polje prima decimalne vrijednosti; vrijednost nula stvara potpuno glatke rubove, a vrijednost jedan stvara nazubljene rubove. [32] Odabirom opcije *remove interior polygons* uklanjaju se poligoni u unutrašnjosti objekta. Kada je ova opcija odabrana, moguće je odabrati i opciju *extrude shards* koja u suštini funkcionira jednako kao kod loma površine. Posljednja nova opcija jest *apply interior material* čijim se odabirom stvara novi zadani materijal pod nazivom *shatterinteriorSG#* i primjenjuje ga se na unutarnje površine fragmenata. Na vanjskim površinama ostaje materijal koji je primijenjen na površini izvornog objekta. Materijali se mogu urediti u prozoru HyperShade gdje je moguće primijeniti različita sjenčenja na unutarnju i vanjsku površinu fragmenta. [32]

5.2.3. Napuknuće

Napuknuće (eng. *crack shatter*) jest efekt koji stvara pukotine koje se šire radijalno od odabrane toče, a može se kreirati samo na poligonalnom objektu otvorene površine. Kao što je vidljivo na slici 5.4, postavke su uglavnom jednake kao u prethodna dva tipa lomljenja.



Slika 5.4. *Prozor za kreiranje napuknuća u Mayi.*

Polje *crack count* podudarno je polju *shard count*. Ostala su polja jednaka onima već opisanim kod loma površine i loma punog tijela. Razliku nalazimo u polju *post operation* čije su opcije jednake kao kod loma površine, osim što nudi još jednu opciju: *cracks on surface*. Ona stvara linije puknuća bez samog lomljenja objekta. Korisna je za brzo pronalaženje obrasca koji se želi koristiti – kreira se lomljenje iz iste točke uz mijenjanje vrijednosti u polju *seed value*. Kada se pronađe željeni obrazac, pripadajući *seed value* upotrijebi se za kreiranje stvarnog loma. Nudi se i jedna potpuno nova opcija, *crak lenght*, kojom se određuje duljina pukotine. Ako upisana vrijednost prelazi udaljenost od odabranog vrha do ruba površine, pukotina će završiti na rubu. [32]

Ovaj detaljan opis postavki i mogućnosti raznih tipova loma koje Mayin ugrađeni alat ima osnova je za njegovo testiranje koje je opisano u odgovarajućim cjelinama šestog poglavlja.

5.3. nCloth lomljenje

Osim ugrađenog efekta lomljenja, postoje i drugi načini za kreiranje loma bez animacije ključnim okvirima (eng. *keyframe animation*), a najčešći je korištenjem Mayinog nCloth modula. nCloth jest brz i stabilan dinamički modul koji koristi sustav povezanih čestica kako bi simulirao širok spektar različitih vrsta dinamičkih poligonalnih površina. [32] Koristi se za simuliranje raznih tipova tkanina, napuhanih balona, raznih elastičnih materijala, deformabilnih objekata i lomljenja. Kreiranje loma pomoću nClotha ipak je više ručnog rada nego što bi se očekivalo. Linije loma određuju se proizvoljno: rezanjem površine pomoću *multi-cut toola*[17] što zahtjeva potpuno poznavanje fizike loma željenog materijala, ako je želja da lom naposljetku izgleda realno. Nakon određivanja linija loma, mreža poligonske površine se triangulira te je potom potrebno naredbom odvojiti komponente. Tek nakon učinjenog ovog koraka, objekt je potrebno pretvoriti u nCloth. [33] Tada je trenutak za dodatna podešavanja karakteristika objekta, npr. materijala, debljine, i sila, npr. gravitacije. Nakon što je objekt spreman za lomljenje, kreira se objekt koji će služiti kao pasivni objekt za sudar (eng. *passive collider*) i translacijom ga se animira do kontakta s objektom. Doduše, u ovom se koraku ipak koristi animacija ključnim okvirima, ali ne za neposredno kreiranje loma, već samo za izazivanje sudara. Ovisno o tipu objekta i tipu pasivnog tijela, moguće je uz lom kreirati i deformaciju. Osnovna ideja, dakle, leži u činjenici da nCloth podržava deformaciju i koliziju, te poštuje naznačene linije loma. Dodatna podešavanja su moguća, ali u suštini postupak je uvijek jednak. Problemi nastaju kada se fragmenti dodiruju – nerijetko se npr. preklapaju, što lom čini nerealnim. [34]

Slika 5.5. *Primjer lomljenja stakla korištenjem nClotha i kugle kao pasivnog tijela za sudar. Screenshot napravljen iz videa www.youtube.com/watch?v=zCXfQFOFWao [dostupno 2. lipnja 2015.]*

[17] Radi se o alatu koji se u starijim verzijama Maye naziva *cut faces tool*. Izmjenom imena nije se izmijenila i namjena alata.

S obzirom na sve rečeno, valja sažeti da se s nClothom daju napraviti zanimljivi i relativno realni lomovi, dostatni za animirane filmove i računalne igre nižeg stupnja realističnosti, a – s obzirom da nema inteligentnog računalnog konstruiranja loma – uspješnost rezultata ovisi isključivo o vještini i kreativnosti grafičara, te njegovom poznavanju fizike materijala. Uz ovaj sažeti uvid u korištenje nClotha za simuliranje loma, primjeri napravljeni ovom tehnikom nalaze se i u izrađenoj videoanimaciji, a dodatno su objašnjeni i u cjelini 6.4.

5.4. Plug-inovi za destrukciju

Uz do sada opisane alate, glavna odgovornost za kreiranje kvalitetnih i raznolikih efekata lomljenja kompleksnih geometrija ipak ostaje na plug-inovima. Plug-inovi su softverske komponente koje se instaliraju uz postojeći softver, dodajući mu određenu značajku koja upotpunjuje ili obogaćuje njegov rad. Za potrebe predmeta ovoga rada odabrano je nekoliko najpopularnijih i najkvalitetnijih plug-inova namijenjenih lomljenju i destrukciji te se u nastavku daje njihov prikaz. Ove softverske nadogradnje uglavnom su relativno skupe što onemogućava njihovo praktično testiranje u sklopu ovoga rada, osim u obliku probnih verzija. Prije toga, daje se uvid u neke od najpopularnijih destrukcijskih plug--inova.

5.4.1. Bullet Physics

S obzirom da je kritika načina na koji Maya manipulira krutim tijelima jedna od najčešćih kritika koje joj se upućuju, Bullet Physics plug-in činio se kao logičan korak za Autodesk. Radi se o softveru otvorenog koda koji je integriran u Mayu 2015, a sada i u 2016. [35] Bullet je vrlo snažan dinamički engine koji uključuje razne opcije, tipa rad s tkaninama i fiziku krpene lutke (eng. *ragdoll physics*)[18], ali glavno područje rada mu je detekcija kolizije i dinamika krutih i mekih tijela. Iako je relativno nov u Mayi, zbog svoje snage, jednostavnosti korištenja i fleksibilnosti, Bullet se koristi već godinama. Kao primjeri, *DreamWorks* je Bullet koristio za animirane filmove „Megamind“ (2010) i „Shrek 4“ (2010), a kroz engine korišten je za igre „Grand Theft Auto V“ (2013/2015[19]) i „DiRT“ (2015).

[18] *Ragdoll physics* jest vrsta proceduralne animacije koja se najčešće koristi kao zamjena za tradicionalnu statičnu animaciju smrti u videoigrama i animiranim filmovima.
[19] GTA V prvo je izdan 2013. godine no samo za konzole PlayStation 3 i Xbox 360. Nadogradnja za nove verzije konzola izašla je 2014. godine, a za PC u travnju 2015.

Slika 5.6. *Lom napravljen s Dynamicom (Michael Baker, SIGGRAPH 2011). [14] [16]*

Osim u Mayu, Bullet dolazi integriran i u Houdini, Cinema 4D, Lighwave, Blender i Carraru, a za 3ds Max postoji samostalan plug-in. Ovakva ga sveprisutnost čini posve nezaobilaznim alatom. Odnos Maye i Bulleta ipak je relativno specifičan. *Disney Animation Studios* razvio je plug-in za Mayu – Dynamica – koji su koristili za animirani film „Bolt" (2008). S obzirom da je studio prestao koristi Mayu (počeli su raditi s Houdinijem zbog njegovog FX sustava) odlučili su Dynamicu donirati Bulletu. Ona je tada postala plug-in otvorenog koda, a Bullet je dobio uvid u potpuno nove tehnike i ideje. Dynamica je specijalizirana za rad s aktivnim, pasivnim i kinematičkim krutim tijelima, kontrolu koreografije objekata te za koliziju konveksnih i konkavnih objekata. Dijelovi njezina kôda sada su sastavni dio Bulleta, a neki od razloga zašto je Bullet danas našao prilično čvrsto mjesto u Mayinu sučelju su upravo oni koje Bullet baštini od Dynamice. [14] Primjer loma na slici 5.6. napravljen je korištenjem Dynamice, a onaj na slici 5.7. kreiran je u Mayi 2015 i Bullet physics plug-inom. Također, model na slici 5.7. korišten je i u izradi videoanimacije, a sažeti postupak opisan je u cjelini 6.5.

Bullet ima diskretnu i kontinuiranu detekciju kolizije, generičko podržavanje konveksnih objekata (kapsula, valjak, stožac, kugla) i nekonveksnih trianguliranih mreža, a pomoću razmještanja strukturne akceleracije, omogućava i dinamičku deformaciju nekonveksnih trianguliranih mreža. [31] Kada je riječ o lomu, uglavnom lomi voronoi

principom, ali postoje i dijelovi kôda koji podržavaju druge tipove loma, uglavnom CSG. S obzirom da se radi o otvorenom kodu, nema profesionalne korisničke podrške, ali dnevno na njemu radi nebrojeno mnogo grafičara i programera, prilagođavajući ga sebi ili novim specifičnim potrebama drugih korisnika. [5]

Slika 5.7. *Lom skulpture u Mayi 2015 korištenjem Bulleta. [36]*

5.4.2. DMM

Digital Molecular Matter – DMM – jest dinamički fizikalni engine koji je razvio Pixelux, a može ga se koristiti i u obliku plug-ina za Mayu i 3ds Max. Za razliku od većine ostalih alata koji simulaciju destrukcije temelje na simulaciji dinamike i kinematike krutih tijela, DMM koristi FEA kako bi simulirao veće setove fizikalnih svojstava. Svojstva objekta modelira u skladu s njegovim željenim karakteristikama te mu omogućuje da se savija i puca u skladu s pritiskom koji se na njega vrši. [34]

DMM je kreiran za potrebe izrade jedne igre u kojoj bi čudovište uništavalo grad. S obzirom da je igra razvijana početkom ovog stoljeća, tehnologija za uništavanje grada nije postojala. Kroz niz istraživanja i isprobavanja raznih tehnika, od volumetrijskog pristupa do geometrijskih modifikacija, utvrdili su da im jedino FEA pruža dovoljno dobre rezultate. Preuzevši tehnologiju koja se do tada koristila za simulacije ponašanja nuklearnih cjevovoda i sl., počeli su kreirati DMM. Iako je Pixeluxov rad postigao izvjesni uspjeh, nedostajala je

glavna iskra. Našli su na rad dr. Jamesa O'Briena i iskoristili njegove ideje – prema riječima Vika Sohala, COO-a tvrtke Pixelux Entertainment:

> „Bio je profesor na Berkeleyu i 1999. godine napisao je rad o deformaciji i lomljenju. Bio je to sjajan rad – no problem je bio što [dok] je njegova simulacija izgledala sjajno, programski je izračun trajao danima. Identičan pristup nije bio održiv za uporabu u filmovima i igrama. Za razbiti npr. pepeljaru trebalo bi dan i nešto izračuna. Takvo nešto nije praktično za korištenje u produkciji. Ideje su ipak bile revolucionarne pa sam ga nazvao, on se javio i sada smo zapeli u tom sjajnom odnosu. I dan danas je jedan od naših savjetnika. Ukratko, zapravo smo uzeli njegovu ideju i samo prilagodili implementaciju.“ [37]

Slika 5.8. *Tom Cruise razbija staklo na najvišem neboderu na svijetu, Burj Khalifa u Dubaiu. Screenshotovi iz filma „Nemoguća misija: Protokol Duh“.*

Iako su postigli dostatnu razinu realističnosti loma, problem je bilo vrijeme potrebno za izračunavanje. S vremenom se brzina odaziva smanjila, a na tome se radi još i danas.

Uspjeh je ipak neporeciv – DMM je korišten u brojnim uspješnim igrama, npr. „Star Wars: The Force Unleashed“ I i II (2008. i 2010), te u popularnim filmovima: npr. korišten je za rušenje drveća i uništavanje helikoptera u „Avataru“ (2009, studio *Weta Digital*); lomljenje stakla u serverskoj prostoriji u „Nemogućoj misiji: Protokol Duh“ (2010, studio *Fuel VFX*); udar vlaka u cigleni zid u „Izvornom kodu“ (2011, studio *MPC*); uništavanje luksuzne jahte u „X-menu: Prva generacija“ (2011, studio *MPC*); rušenje tornja, drveća i zidova u „Sherlocku Holemesu: Igra sjena“ (2011, studio *MPC*) itd. [37] Spomenute scene u „Avataru“ i „Nemogućoj misiji“ napravljene su upravo u Mayi uz DMM plug-in. Na slici 5.8. prikazana je spomenuta scena iz „Nemoguće misije“, a na slici 5.9. iz „X-mena“.

Osnovne tehnike na kojima počiva DMM su FEA i tetraedarizacija[20], što je pravo osvježenje za Mayin sustav simulacije krutih tijela. Sukladno tome, DMM je došao ugrađen uz Mayu 2012 i 2013, ali samo u obliku demo verzije. Iako je time plug-in postao korišteniji, njegova praktična uporaba ipak je bila moguća samo uz kupnju pune verzije – naime, demo verzija bila je ograničena za uporabu na objektima s najviše 2500 tetraedara. Cijena pune verzije jest 500 USD.

Kako bi se u Mayi na objektu izazvao efekt lomljenja korištenjem DMM-a, objekt nije potrebno unaprijed rezati niti mu ni na koji način određivati linije pucanja, već se objekt odmah pretvara u DMM materijal. Ima vlastiti vodič s uputama (eng. *user guide*) i zapravo je relativno brz kod *playbacka* pa nije potrebno raditi kopije između probnih intervencija na objektu. [34] Ukratko, alat je vrlo intuitivan i jednostavan, a brz i koristan. Kao i svaki, ima svoje mane, ali neporecivo je da se ističe u moru sličnih alata – po inovativnosti, popularnosti i kvaliteti.

Slika 5.9. *Uništavanje luksuzne jahte u X-menu. [38]*

[20] O tome je već nešto rečeno u cjelinama 3.1.1.1. i 3.1.3.

5.4.3. Fracture FX

Fracture FX jest proceduralni[21] plug-in za Mayu namijenjen za različite vrste destrukcije. Sučelje mu je na prvi pogled relativno nepregledno, ali ništa što se ne može savladati uz proučavanje navigacije kroz rad. Podržava višestruke lomove, detekciju kolizije te odlično surađuje s Mayinim dinamičkim poljima i mnogim ostalim plug-inovima. [41] Slika 5.10. prikazuje lomljenje zida u Mayi pomoću Fracture FX-a, uz dodatno kreiranje prašine s plug-inom FumeFX, dok na slici 5.11. nalazimo primjere reklamnih slika napravljenih za kampanju komercijalne promocije ovog plug-ina.

Slika 5.10. *Lomljenje zida. [39]*

Kada je u pitanju lom, Fracture FX podržava voronoi i nevoronoi lomljenje. Nudi četiri obrasca voronoi loma: *uniform*, *poison*, *radial* i *cluster*. Nudi nekoliko tehnika nevoroni lomljenja, npr. kreiranje obrasca kontinuiranim linijama, iteracijama ili bojom. [40]

Bez obzira na odabranu tehniku, alat je kvalitetan i nudi dobru kontrolu nad lomom i ponašanjem objekta. Plug-in košta 400 USD za komercijalnu verziju i 200 USD za obrazovnu licencu (27. ožujka 2015). Osrednja cijena i osrednja popularnost za alat koji je zapravo iznad

[21] Kada je riječ o ovakvom softveru, proceduralnost u primjeni znači da alat dopušta mijenjanje postavki, modula i karakteristika jednom postavljene animacije. Konkretno, ako se postavi scena za lom i odredi da kamen razbija površinu radijalno, tip loma može se promijeniti nakon što se pogleda animacija, bez ikakvog dodatnog prilagođavanja.

prosjeka – izlazi iz okvira prosječnih Mayinih destrukcijskih „voronoi i eksplozija“ plug-inova.

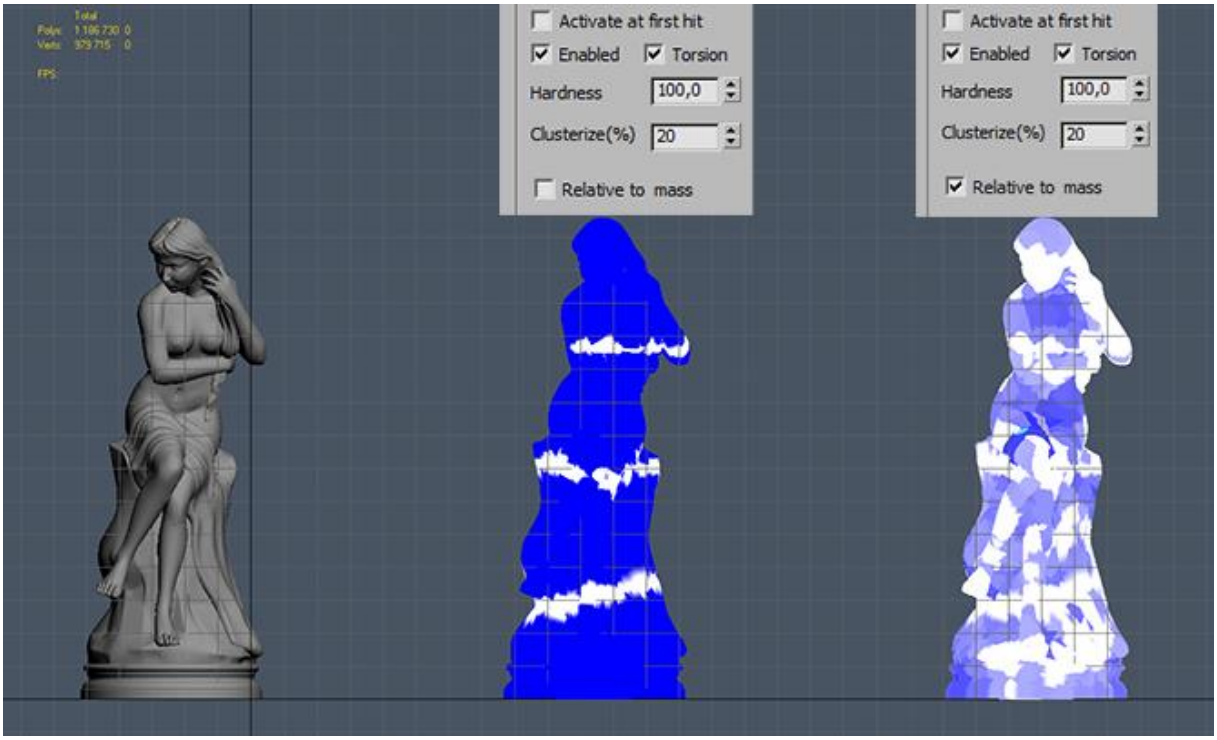
Slika 5.11. *Implementacija lomljenja stakla Fracture FX-om u fotografije. [40]*

5.4.4. PullDownIt

Razvoj alata PullDownIt počeo je 2009. godine za potrebe posebnog projekta: realistično digitalno uništavanje španjolskog akvadukta Segovia. Nakon nekoliko mjeseci rada, scena je kreirana i objavljena na internetu. [42] S obzirom da je doživio veliki uspjeh u svijetu grafike i popularnost na društvenim mrežama, oko projekta je osnovana tvrtka „Thinkinetic“ koja i danas razvija PullDownIt. Trenutno je dostupan kao plug-in za Mayu i 3ds Max na svim platformama. Cijena licence je od 395 EUR za jedno računalo i jedan sustav (1. travnja 2015).

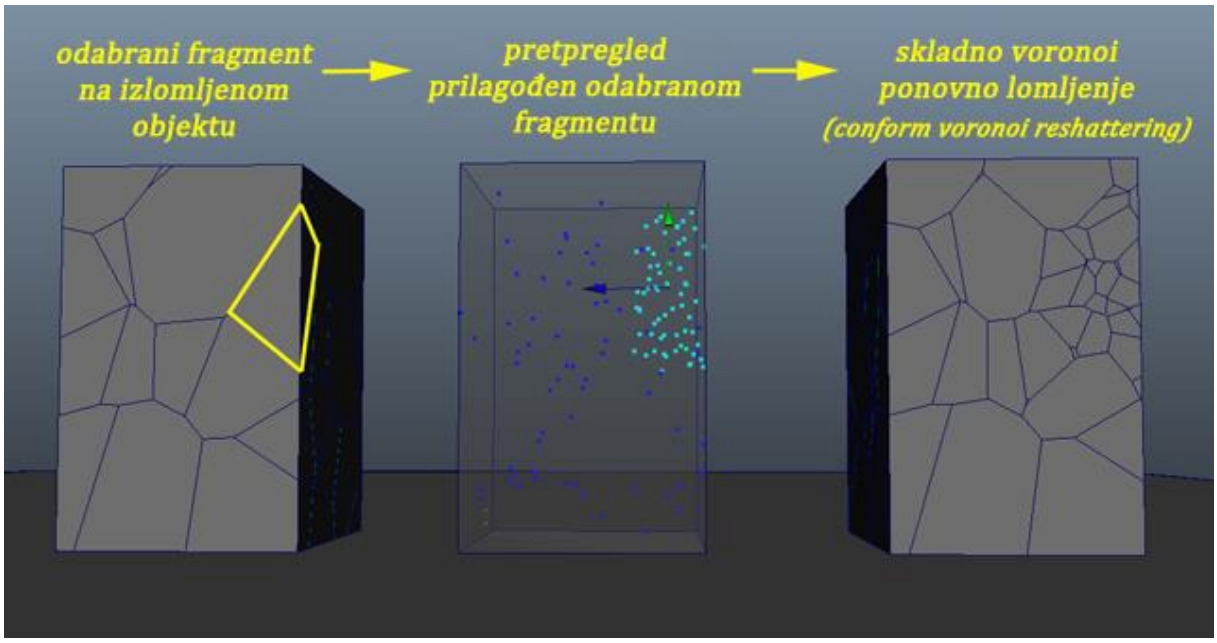
Slika 5.12. *Uništavanje raznih materijala na sceni korištenjem PullDownIta u Mayi. Izvor slike: www.cgmeetup.net/home/destroying-scenery-pulldownit-3-maya/ [dostupno 4. svibnja 2015.]*

PullDownIt jedan je od rijetkih plug-inova koji nisu zapravo derivati istoimenih engina, kao recimo DMM i Bullet, već ima vlastite algoritme koji su razvijeni upravo za samo jednu svrhu: lom, odnosno destrukciju 3D objekata. Također, ovaj plug-in bio je prvi koji je mogao dodati hrapavost voronoi fragmentima. Tu su mogućnost razvili 2012. godine, a rezultat je bilo značajno povećanje realističnosti loma. Tada je dodana i opcija da se pri kreiranju loma u obzir uzme distribucija mase objekta čiji je primjer prikazan na slici 5.13. U skladu s time, objekt bi se različito – manje realistično – lomio bez odabira te opcije, dok bi njezino uključivanje sugeriralo realnija mjesta loma i fragmenata. [42]



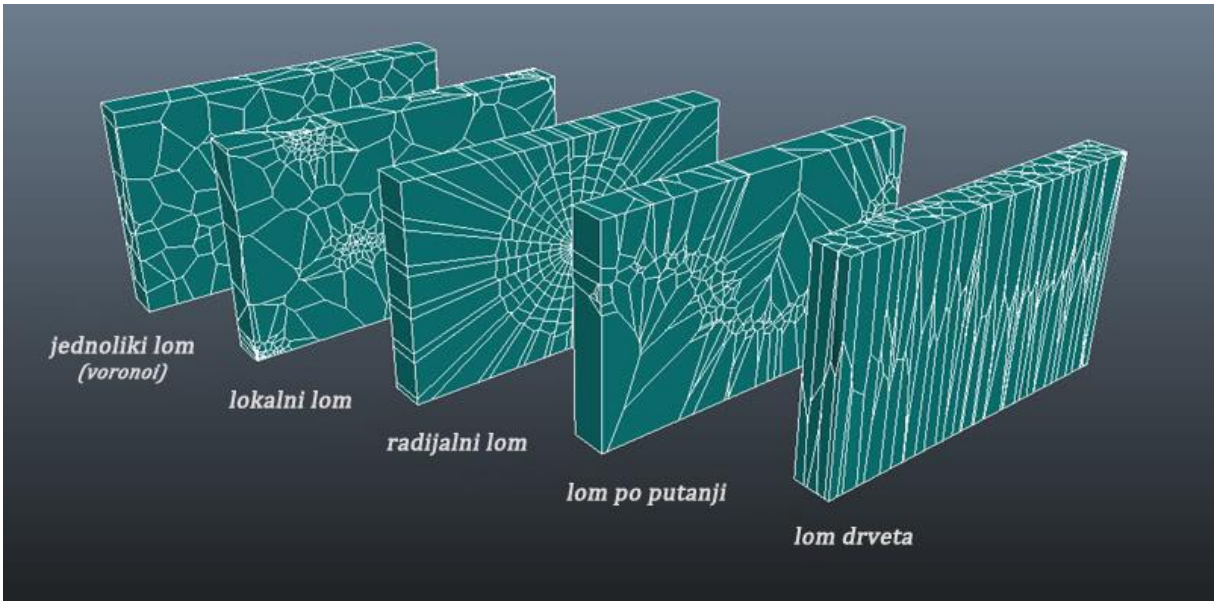
Slika 5.13. *Lomljenje objekta ovisno uzima li se u obzir distribucija njegove mase ili ne. [43]*

Godine 2014. uveli su još neke novitete, kao što je *conform voronoi reshattering* prikazan na slici 5.14. – uklanjanje nerealističnih ravnih granica između fragmenata i ponovno lomljenje modela u odabranom području. Sve to se moglo dodatno kombinirati s dodavanjem neravnina i hrapavosti na rubovima. Alat pruža i mogućnost dodavanja sekundarnih lomova, linija napuknuća i sličnih popratnih elemenata koji lom čine realističnijim, a alat kvalitetnijim.



Slika 5.14. *Kreiranje skladnog voronoi ponovnog lomljenja.*
Vlastiti prikaz temeljen na literaturi. [42]

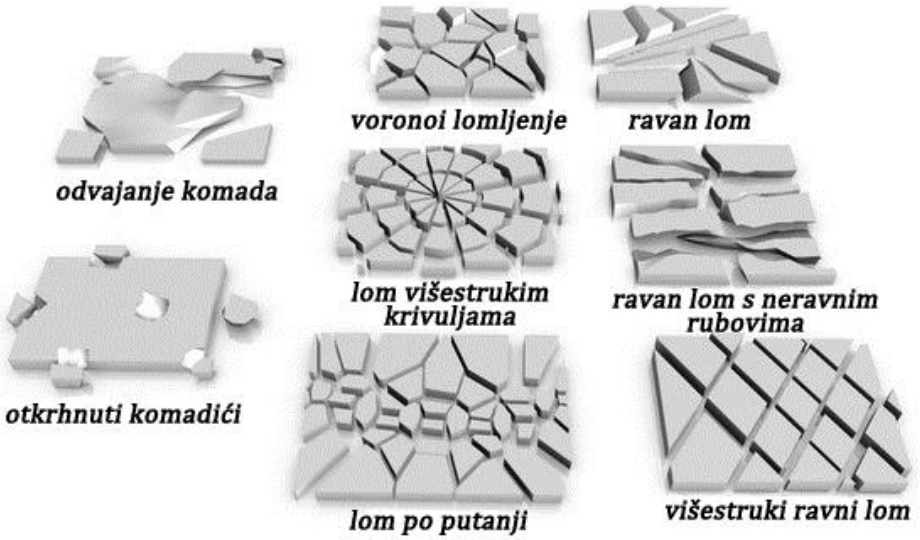
Ono što također valja napomenuti jest da PullDownIt nije dinamički alat što znači da je lomljenje unaprijed jedini način loma – lom se ne simulira dinamički i pri simulaciji neće nastati nikakvi fragmenti koji nisu unaprijed definirani. Iako ga ova karakteristika ne čini prevelikim tehničkim pionirom, i u startu ga diskvalificira kao alat za rad na grafici u stvarnom vremenu, alat je dovoljno opremljen za vlastito kreiranje prijelomnih linija različitih tipova i materijala te u tom pogledu ne zahtjeva previše ručnoga rada. Osnovni obrasci loma koje PullDownIt nudi prikazani su na slici 5.15.



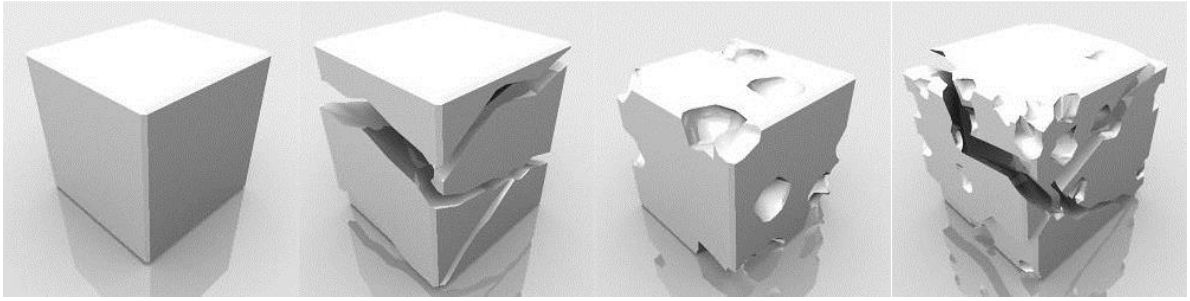
Slika 5.15. *Osnovni obrasci loma u PullDownItu. [44]*

5.4.5. Skripte

S obzirom da Maya poznaje programske jezike raznih razina (C++, MEL, Python) i dopušta korištenje njihovih naredbi u za to određenom sučelju, osim korištenjem plug-inova, razne je efekte moguće postići učitavanjem programskih skripta. One su najčešće značajno jeftinije, ponekad i besplatne. Neke se sastoje od tek nekoliko linija kôda i primjenjuju se za jednostavne radnje kao zaobljavanje oštrih rubova na mnogokutima, a neke su nepregledni nizovi kôda te se koriste i za najkompleksnije zadatke. Među ovim kompleksnijima nalazi se i pitanje loma, a kao primjer daje se skripta „Ninja Fracture“. Košta 40 USD (2. svibnja 2015), ima jednostavno, ali korisno sučelje i podržava različite tipove loma. Poznaje voronoi dijagrame, boolean operatore i dinamiku. [45] Kao što je vidljivo na slici 5.16, moguće je odabrati različite obrasce loma. Tome treba prigovoriti jedino da lom po putanji ovdje zapravo nije zasebni obrazac niti algoritam (kao što je npr. kod PullDownIta), već se radi o vođenom voronoi lomljenju. Primjer kombiniranih odabranih obrazaca nalazi se na slici 5.17.



Slika 5.16. *Mogući obrasci loma u Ninji Fracture. [45]*



Slika 5.17. *Prikaz lomljenja objekta kombinacijom obrazaca otkrhnutih komadića i ravnog loma s neravnim rubovima. [45]*

5.4.6. Usporedba

Iako bi se za svaku od opisanih nadogradnji mogla pronaći pojedina situacija u kojoj je baš ona najbolji odabir, tablica 5.2. daje sažeti usporedni prikaz svih njih.

		BULLET	DMM	FRACTURE FX	PULL DOWNIT	NINJA
SUČELJE	preglednost	+	+	-	+	+
	količina postavki	detaljno	detaljno	sažeto	sažeto	sažeto
	navigacija	jasna	jasna	nejasna	jasna	nejasna
UPUTE	korisnička podrška/*user guide*	-	+	+	+	+/-
	tutorijali	+	+	+	+	-
UPORABA		široka	spec.	uska spec.	uska spec.	uska spec.
LOM	tip/tehnika loma	voronoi, CSG	FEA, tetrae-darizacija	voronoi, nevoronoi	voronoi	voronoi, CSG
	sekundarni alati	+	+	-	+	-
	obrasci	-		4	5	8
DINAMIČNOST		+	+	-	-	+/-
PROCEDURALNOST		+/-	+/-	+	-	-
PROŠIRIVOST/ PRILAGODLJIVOST		+	-	-	-	+
PUNA CIJENA		besplatan	500 USD	400 USD	395 EUR	40 USD

Tablica 5.2. *Usporedni prikaz opisanih nadogradnji.*

U prvoj se kategoriji uspoređuju sučelja: njihova preglednost, količina njihovih postavki te jasnoća navigacije. U drugoj se kategoriji uspoređuje dostupnost uputa u obliku službene korisničke podrške, odnosno službenog vodiča za korisnike te u obliku raznih tutorijala. Oznaka +/- u ovoj kategoriji kod Ninje Fracture skripe znači da, iako nema korisničke podrške u uskom, profesionalnom smislu riječi, autor(i) kôda primaju sva pitanja i

prijedloge te na njih javno odgovaraju, što se donekle može shvatiti službenim uputama. Treća kategorija govori o (ne)specijaliziranosti njihove uporabe. Četvrta kategorija bilježi informacije o lomu: njegovom tipu, odnosno tehnici, o postojanju sekundarnih alata (za korekciju, uređivanje, prilagođavanje i sl. loma u najširem smislu) te o broju ponuđenih obrazaca. Kod DMM plug-ina u ovoj kategoriji ćelija je zacrnjena jer on ne posjeduje obrasce loma, no ne zato što u alatu nedostaju, već zato što radi FEA procedurom koja obrasce uopće ne zahtjeva. Peta i šesta kategorija bilježe jesu li nadogradnje dinamične, odnosno proceduralne. Oznake +/- kod Bulleta i DMM-a označavaju relativnu proceduralnost budući da ti alati nude i neke tehnike koje su proceduralne i neke koje nisu. Sedma kategorija donosi informaciju jesu li alati na bilo koji način prilagodljivi, bilo nadogradnjama bilo samostalnim programiranjem. I, naposljetku, osma kategorija sadrži informaciju o nepovlaštenoj cijeni pune verzije nadogradnje.

5.5. Utjecaj na performanse

Nezaobilazna tema, kada je riječ o destrukciji, odnosno efektu loma, jest utjecaj na računalne performanse. Pri opisu određenih plug-inova spomenuta je njihova brzina, a to se najviše odnosi na alate koji su namijenjeni (i) za simulaciju u stvarnom vremenu, ili su barem prvotno tako zamišljeni. Pojam brzine za klasične 3D programe ipak je vrlo relativan. Kada se radi o dinamici, onda je gotovo pa neodređen. Efekt loma spada u najzahtjevnije procese, posebice kada se radi o dinamičkoj i proceduralnoj dinamici. Svaki pretpregled animacije traži zahtjevno izračunavanje svakog pojedinog okvira. Uzmemo li kao primjer jednostavnu animaciju napravljenu u Mayi bez korištenja plug-inova, npr. kugla pada na pod i razbija se u tri komada, bez posebnih efekata, deformacija i sekundarnih lomova, njezin će pretpregled na prosječnom računalu namijenjenom za 3D grafiku iznositi oko pola sata, a renderiranje npr. dvostruko duže.

Faktori koji utječu na brzinu oba procesa, pretpregled animacije i renderiranje, brojni su i često ovise jedni o drugima. Recimo, oba će procesa duže trajati ako je objektu dodijeljena složenija tekstura, no to će značajnije utjecati na renderiranje, njegovo trajanje i/ili kvalitetu, ovisno o alatu koji se koristi i parametrima koji se zadaju, a utjecaj na pretpregled bit će manje značajan. S druge strane, odabir tehnike loma i njegovih parametara imat će veći učinak na produženje trajanja izračuna za pretpregled, nego na trajanje renderiranja. Ako se želi manipulirati vremenom za pretpregled, treba se pažljivo birati postavke i, kada je riječ o Mayinom ugrađenom alatu za efekt loma, držati se ispitivanja najjednostavnijih scena.

Također, tehnike koje nisu dinamičke, ili su dinamičke u manjem stupnju, rezultiraju kraćim vremenom izračuna što naposljetku znači i kraće vrijeme odaziva za svaki pojedini okvir pri pretpregledu. Dakle, tehnike koje uključuju rezanje objekta prije simulacije loma, vrte se brže od potpuno dinamičkih tehnika koje samostalno izračunavaju čitav lom. Važno je, također, napomenuti kako je pri pretpregledu nužno animaciju pustiti okvir po okvir jer softver treba izračunati svaki pojedini okvir za sebe. Puštanje pretpregleda u stvarnom vremenu, ili u nekoj od drugih opcija, uglavnom će rezultirati fatalnom pogreškom i rušenjem programa. Općenito, kada je riječ o Mayinu ugrađenu alatu za efekt lomljenja, velik broj kombinacija postavki za lom mogu rezultirati rušenjem. Triangulacija površine i zaglađivanje rubova kod fragmenata dovodi najmanje do udvostručenja vremena potrebnog za obradu, a ako je objekt podvrgnut djelovanju nekih dinamičkih sila ili bilo kakvoj animaciji, vrijeme se dodatno produžuje.

Kod loma kreiranog pomoću nClotha također se pažljivim odabirom postavki može utjecati na smanjenje vremena potrebnog za obradu. Prije toga, spomenimo samo da, kada je u pitanju brzina pretpregleda animacije, nClothom se lomi mnogo brže no ugrađenim alatom za lom punog tijela, ali malo sporije nego ugrađenim alatima za lom površine i napuknuće. Ako se lomu nClothom posveti dodatna pažnja, može ga se učiniti bržim i od njih, čak i kod nešto kompleksnijih geometrija. Savjete za ubrzanje daje direktno Duncan Brinsmead, Mayin programer najbolje poznat kao kreator raznih alata za simulaciju realističnog 3D okoliša i kao autor Mayinih *Paint Effectsa*.

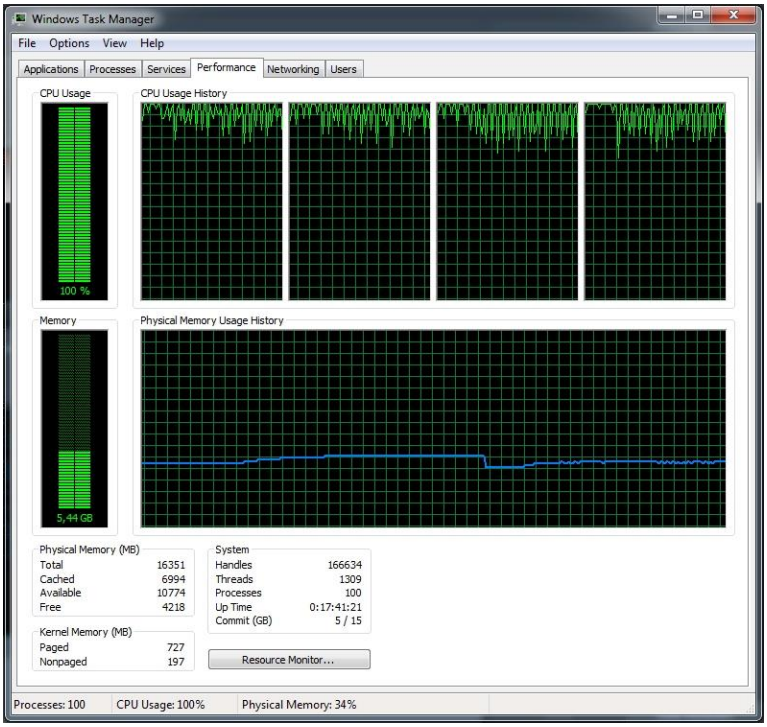
Kako bi se smanjilo vrijeme odaziva pri kreiranju loma nClothom, Brinsmead savjetuje nekoliko stvari. Ručnim odabirom samo točaka, rubova ili poligona pri kreiranju ograničenja loma, lomne linije stvarat će se samo na odabranim dijelovima i na taj način lomljenje učiniti bržim. [46] Interni kôd za odvajanje komponenata unaprijed definirano lomi po točkama (eng. *vertex*), a ne po rubovima (eng. *edge*) tako da se lomljenjem dobivaju vrlo grubi i oštri rubovi. To se može izbjeći pažljivim modeliranjem: naprosto se model osmisli tako da željena mjesta lomu budu ujedno i granični rubovi. Time se dobivaju prirodnije lomne linije i smanjuje se vrijeme potrebno za izračun loma.

Ako se želi dobiti velike krute fragmente, boje je raditi s više odvojenih objekata od kojih je svaki zaseban nCloth. Manje je vrijeme potrebno za obradu više manjih objekata, nego za jedan vrlo velik. [46] Zasebne objekte potrebno je međusobno povezati *component to component* metodom uz *glueStrength* i metodu zavarivanja (eng. *weld mehod*). Zatim se kombiniraju, spajaju točke, zaglađuju rubovi i eventualno istisnu niže od nCloth čvorova. To omogućuje korištenje svojstva krutosti na nCloth čvorovima, što bi se u suprotnom postizalo postavljanjem visoke razine savijanja/rastezanja materijala, no to je manje učinkovita tehnika.

Može se u nCloth čvorovima isključiti kolizija, s obzirom da se interno odmah radi o krutim tijelima. [46] Isključena kolizija uvijek znači kraće vrijeme potrebno za izračun. Ako se koristi jedan nCloth čvor za sve objekte, onda kolizija mora ostati uključena jer bi se u protivnom fragmenti mogli preklapati.

U čvoru za namještanje ograničenja postoji polje za otpornost pri savijanju (eng. *bend stiffnes*) koje se može koristiti za kreiranje loma u tipu razbijanja stakla, no to zahtjeva da za rub postoje nComponents. Ako se koriste standardni lomljivi objekti s definiranim rubnim dijelovima, korištenje polja za otpornost pri savijanju rezultira ekstremno dugim vremenom potrebnim za odaziv, a kod kompleksnih geometrija često dolazi i do rušenja programa. Taj problem još nije riješen, javlja se i kod lomljenja nClothom i kod loma punog tijela ugrađenim alatom, a prema Brinsmeadovim riječima, na njemu se intenzivno radi. Rezultat toga rada očekuje se u najranije u Mayi 2017. [46]

Kada je riječ o renderiranju, najbolja opcija jest kombinirano CPU i GPU renderiranje, dok se samostalno CPU renderiranje pokazuje jednako uspješnim kao GPU u slučajevima kada su modeli relativno jednostavni i nemaju zahtjevne teksture. Kod zahtjevnih tekstura, GPU renderiranje pokazuje se kvalitetnijim od CPU, ali samo u pogledu kvalitete slike i interpretacije materijala i boja. Takva je situacija očekivana s obzirom da se, kada je u pitanju efekt loma, problem svodi na izračun lomnih linija i slične uglavnom matematičke probleme.



Slika 5.18. *CPU i RAM opterećenje pri renderiranju najjednostavnijeg lomljenja kugle.*

6. Primjena – izrada videoanimacije

S obzirom na to da je područje računalnog lomljenja i destrukcije izuzetno zanimljivo, znatiželja je bila glavni razlog za odabir ove teme. Osnovna namjera bila je stjecanje novih vještina i proširenja znanja o dinamici dodatnim saznanjima vezanim uz dinamiku krutih tijela – onaj aspekt Mayina rada o kojemu se uglavnom manje govori. Gledavši tutortijale i *youtube* videe rad je zamišljen kao spoj kvalitetnog modeliranja i sveopće destrukcije: plan je bio modelirati ulicu, njezine građevine, cestu, raslinje, i sl. te u njezinu skladnom i snažnom razaranju pokazati razne tipove loma, one realistične i manje realistične, one dinamičke i manje dinamičke, lomove ovisno o materijalu, ovisno o mjestu udara, ovisno o tipu detekcije kolizije itd. Ishod rada ispao je dijametralno suprotan - što zbog inicijalnog nepoznavanja ugrađenog alata, što zbog nemogućnosti kupovanja raznih skupih plug-inova.

Prva stepenica spoticanja bilo je saznanje da se alat za efekt loma koji je ugrađen u Mayu nikako ne može koristiti na kompleksnijim modelima – program se znao rušiti i na animiranim primitivima kojima je zadano lomljenje na šest ili sedam fragmenata. Ako bi se uz to još nastali fragmenti htjeli definirati kao kruta tijela s uključenom detekcijom kolizije - a ne obični oblici kao što je unaprijed postavljeno – program bi javljao fatalnu pogrešku i rušio se u 80% situacija. U slučajevima kada se ne bi rušio, pretpregled animacije trajao bi satima što je rad učinilo nemogućim – samo za promatranje ponašanja objekta pri izmjeni kombinacija osnovnih postavki doslovno su trebali tjedni. Kada su u čitavu kombinaciju još uvedena dinamička polja, situacija se dodatno komplicirala. Za određene simulacije nužno je korištenja pasivnih tijela za sudar (eng. *passiv colliders*) i pasivnih krutih tijela (eng. *passiv rigid bodys*), no oni stvaraju problem jer na njih dinamička polja ne utječu. Neprirodno nestaju iz okvira scene ili, još neprirodnije, ostaju lebdjeti u zraku. Također, u određenim situacijama razlomljeni bi objekti neobjašnjivo uranjali u druge objekte ili tonuli kroz pod iako bi oni bili definirani kao pasivna tijela. Ovakve, i razne druge situacije nužno su zahtijevale razne intervencije, često u obliku animacije ključnim okvirima, što se direktno kosilo s predmetom rada – cilj je bio pokazati da Mayin ugrađen alat znatno olakšava animaciju loma u odnosu na tradicionalno animiranje, a potpomaganje animiranjem ključnim okvirima nije to činilo. S ovakvim ukratko danim uvodom u proces izrade praktičnog rada, može se krenuti dalje: pružiti jasniji prikaz ovog – uglavnom eksperimentalnog i istraživačkog – rada te njegovih rezultata. Osim Mayinog ugrađenog alata korišteni su i dostupni plug-inovi. Rad s njima bio je neusporedivo lakši no rad s Mayinim efektom.

Kako bi vrijeme koje je bilo potrebno za određene procese te sveukupni rezultati bili usporedivi, daje se konfiguracija računala na kojem su rađena sva ispitivanja i svi primjeri. S obzirom da sustav treba skoru nadogradnju, relevantne komponente su overklokane.

Specifikacije računala su:

softver:	OS Windows 7, 64-bitni; Autodesk Maya 2016 (oboje na SSD SATA III, Agility 3)
grafička kartica:	GTX560-Ti, TwinFroz II, 2GB VRAM, DirectX11 (overklokano)
CPU:	s1155 Intel Core i5 2500K 3.3GHz (overklokano)
RAM:	4 x 4GB G.Skill DDR3 1600 MHz (overklokano)

6.1. Odabir alata

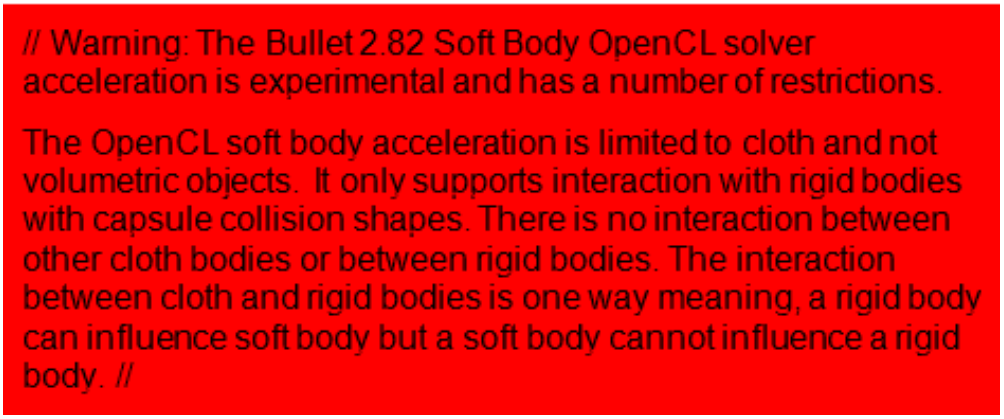
Osim Mayina ugrađenog alata za kreiranje efekta loma te lomljenja pomoću nClotha, isproban je i rad nekih od plug-inova. Skripta opisana u cjelini 5.4.5. nije dorađena kako bi funkcionirala u Mayi 2016 tako da njezino testiranje nije bilo moguće.

Kao što je navedeno u cjelini 5.4.1., Bullet Physics dolazi unaprijed dostupan u Mayi 2016, no potrebno je omogućiti njegovo korištenje klikom na izbornik *Window* > *Settings/Preferences* > *Plug-in Manager*. Kada se otvori prozor *Plug-in Manager*, u njemu je potrebno odobriti učitavanje za bullet.mll. Nakon učitavanja, Bullet se dodaje u Mayin glavni izbornik, a Bulletove opcije na *shelf* kao što je prikazano na slici 6.1.

Slika 6.1. *Bulleotove opcije na shelfu.*

Iako besplatan, njegov rad u Mayi ipak je ograničen u područjima koja su još eksperimentalna. Određene su restrikcije bile relevantne za predmet ovog rada budući da su utjecala na kreiranje loma i odabir objekata u koliziji. Jedna od takvih restrikcija prikazana je na slici 6.2., a rezultirala je time da je odabrano tijelo kojim se objekt razbija univerzalno kugla, odnosno maksimalno konveksno tijelo (kako bi se izbjegao problem konkavnosti). Također, potaknuto ovom restrikcijom, odlučeno je da će se Bulletom razbijati samo kruta

tijela – što zbog toga što je Bulletu upravo to najjača strana, a što zbog toga što je Mayin rad s mekim tijelima sam za sebe već daleko napredniji od Bulletovog.



Slika 6.2. *Jedna od Bulletovih restrikcija u Mayi.*

Usput rečeno, Bullet 2.83 nedavno je pušten u javnost, a i prezentiran je na ovogodišnjem SIGGRAPH-u: 11. kolovoza ove godine na 90-minutom predavanju predstavljene su njegove razne značajke i poboljšanja, a neka od njih obećavaju i jačanje njegove uloge u Mayi 2017.

Od ostalih plug-inova opisanih u cjelini 5.4., DMM se u trenutku izrade rada mogao instalirati na Mayu 2016, no nije funkcionirao i stoga nije bilo moguće testirati njegov rad. Fracture FX i PullDownIt imali su izdane inačice za Mayu 2016, no s obzirom da se radi o komercijalnim softverima, bilo je moguće testirati samo probne verzije.

Fracture FX svoju je probnu verziju ovako ograničio:

- Mreže od *break selected* i pod *fragmentGroup* ne mogu se uređivati, mijenjati, renderirati niti izvoziti u druge formate. U svrhu proučavanja fragmenata moguće je jedino razdvojiti mrežu (isključivo putem izbornika za lom) u više manjih mreža (za koje opet vrijede ranije navedene restrikcije).
- Datoteke se ne mogu spremati.
- Lomljenje je ograničeno na petsto voronoi točaka i pet lomnih linija.
- Simulator je ograničen na dvjesto okvira.
- Onemogućena je izmjena selekcije na unutarnje/vanjske površine.
- Može se napravit *bake to keys*, ali ne na disk.
- Međukoraci kod *bake to keys* nisu dostupni, odnosno nema kontrole nad postupkom.

- Probna verzija ne uključuje API.
- Probna verzija ne uključuje izmjenjive MEL skripte.

Također, prema riječima vlasnika programa, čisto da bi korisnike probnih verzija držali na oprezu – svakih 15-ak minuta prikazuje se skočni prozor s upozorenjem o korištenju probne verzije. Zaključno, s obzirom da se u probnoj verziji gotovo uopće ne može utjecati na lom, niti ga renderirati ili spremiti, pomoću ovog plug-ina nije bilo moguće napraviti primjer u videoanimaciji.

Posljednji od opisanih plug-inova je PullDownIt koji je svoju probnu verziju ograničio kako slijedi:

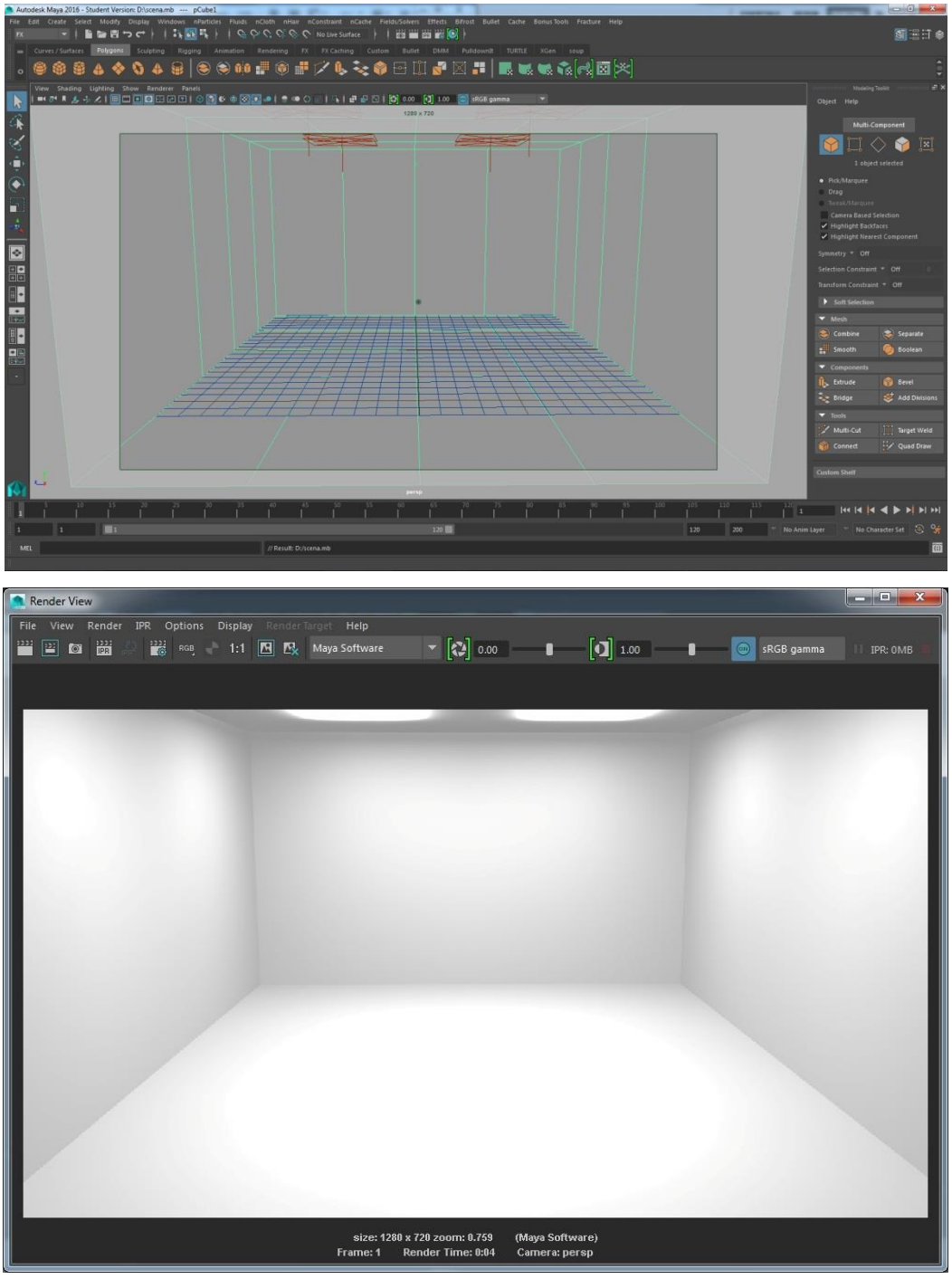
- Onemogućeno je korištenje objektnih dinamičkih polja.
- Maksimalan broj fragmenata u jednom lomu jest 64.
- Maksimalan broj fragmenata po objektu jest 200.
- Na sceni može biti samo jedan objekt koji se lomi.
- Maksimalan broj klastera po tijelu koje se lomi jest tri.
- Ako se na sceni nalazi više od 300 objekata, brojeći i fragmente, maksimalan broj okvira za simulaciju je 100.

Iako prilično ograničavajući, ovakvi su uvjeti ipak dopustili da se pomoću ovog plug-ina napravi nekoliko primjera za videoanimaciju. Tijela su morala biti jednostavna i nije bilo moguće koristiti obrasce koji zahtijevaju veći broj fragmenata, ali se način loma i njegova relativna realističnost ipak mogla testirati. PullDownItove opcije, kao i Bulletove, nalaze se na *shelfu* i prikazane su na slici 6.3.

Slika 6.3. *PullDownItove opcije na shelfu.*

6.2. Scenografija

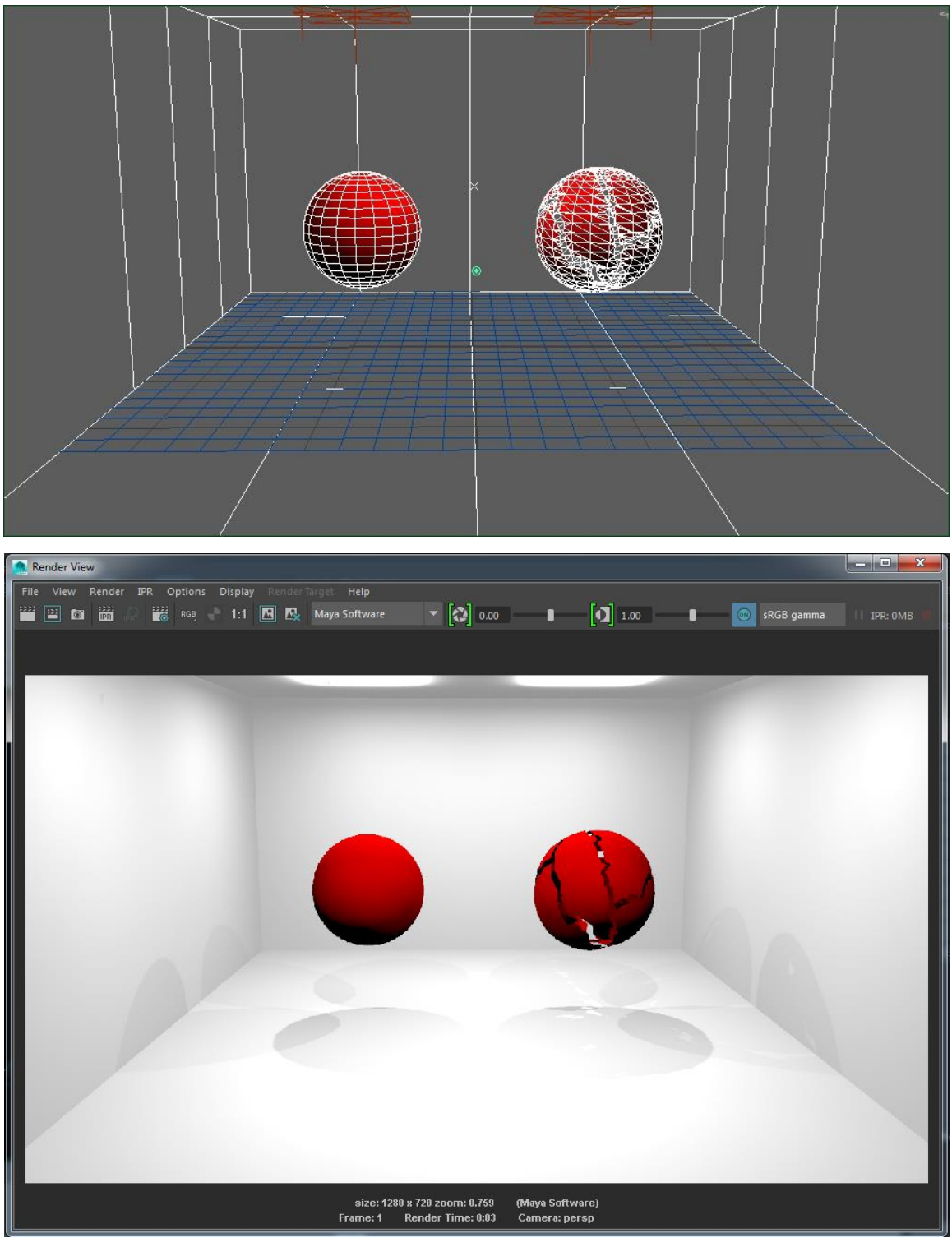
S obzirom na to da se rad okrenuo u smjeru testiranja raznih alata pri čemu nije bilo moguće koristiti gotovo ikakve imalo kompleksnije modele, kreiran je prostor u kojem će testiranja biti prikazana. Prikaz prostora s namještenim kutom kamere i pripadajućom rasvjetom nalazi se na slici 6.4.



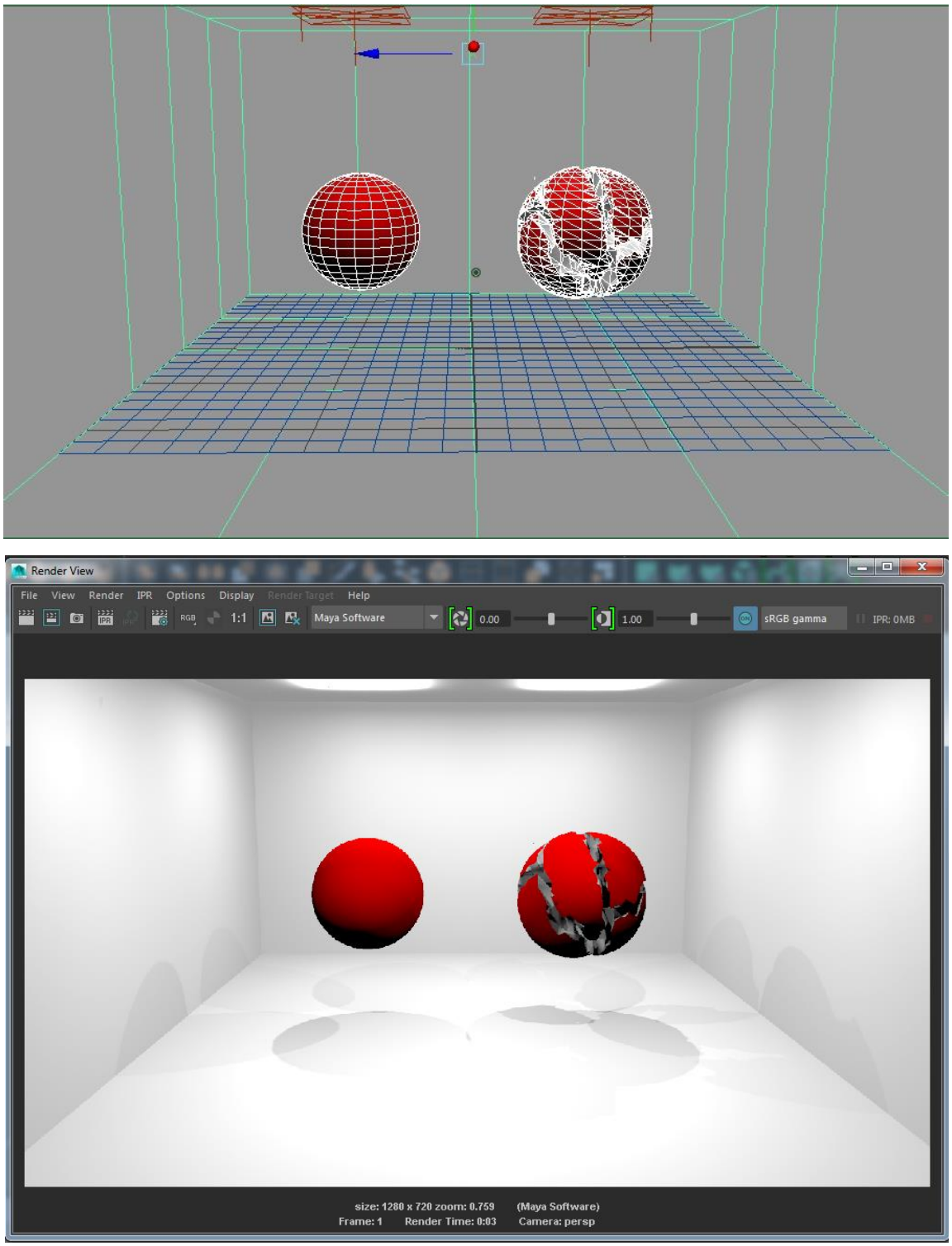
Slika 6.4. *Prikazi scenografije u Mayi i u render prozoru.*

6.3. Lomljenje pomoću ugrađenog alata

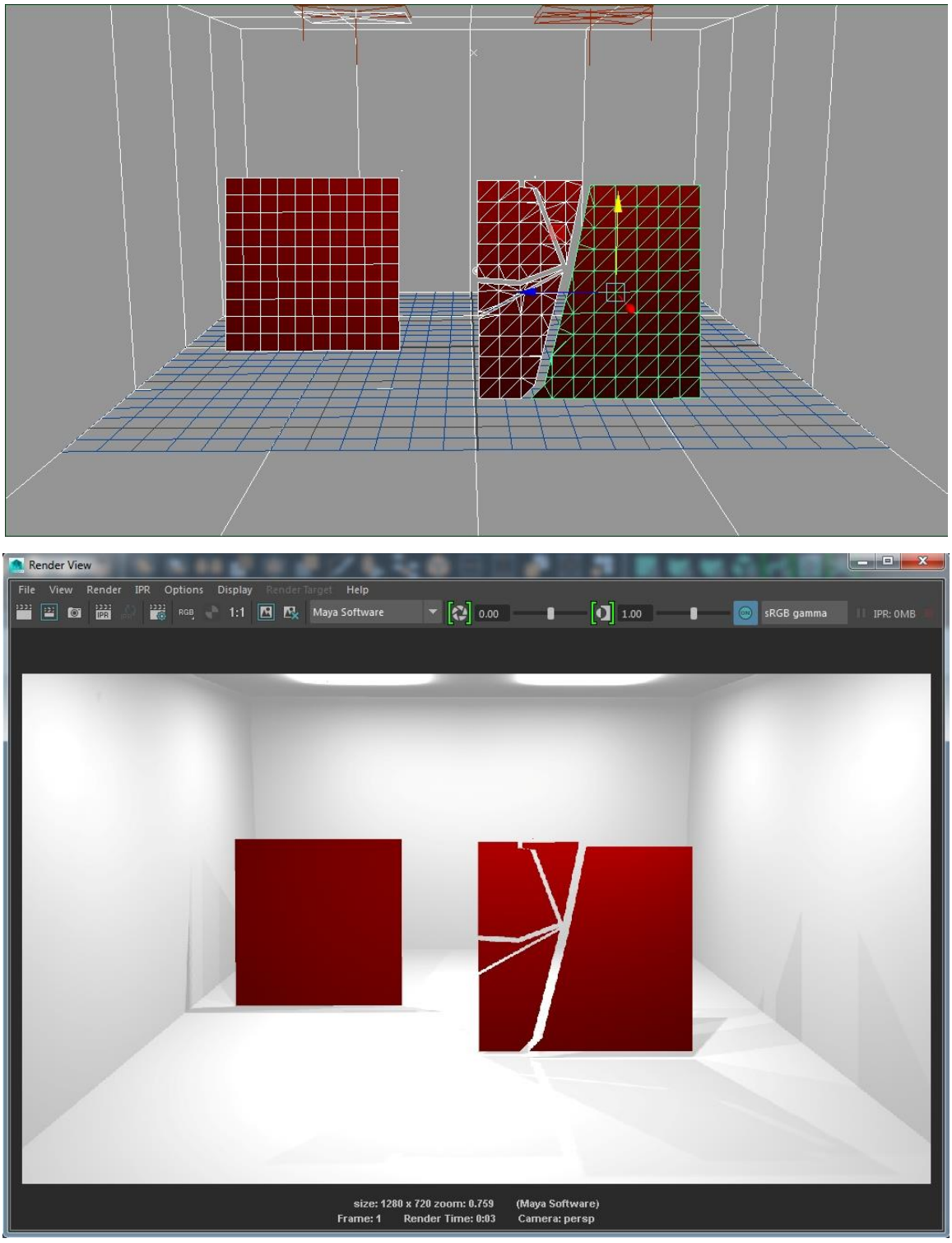
Kao što je detaljno opisano u cjelini 5.2., Mayin ugrađeni alat nudi tri tipa loma: jedan tijelo lomi kao da je šuplje, drugi kao da je puno, a treći lomi otvorene površine. Primjeri tih lomova nalaze se na slikama 6.5., 6.6. i 6.7.



Slika 6.5. *Lom površine prikazan na kugli (u Mayi i u render prozoru).*



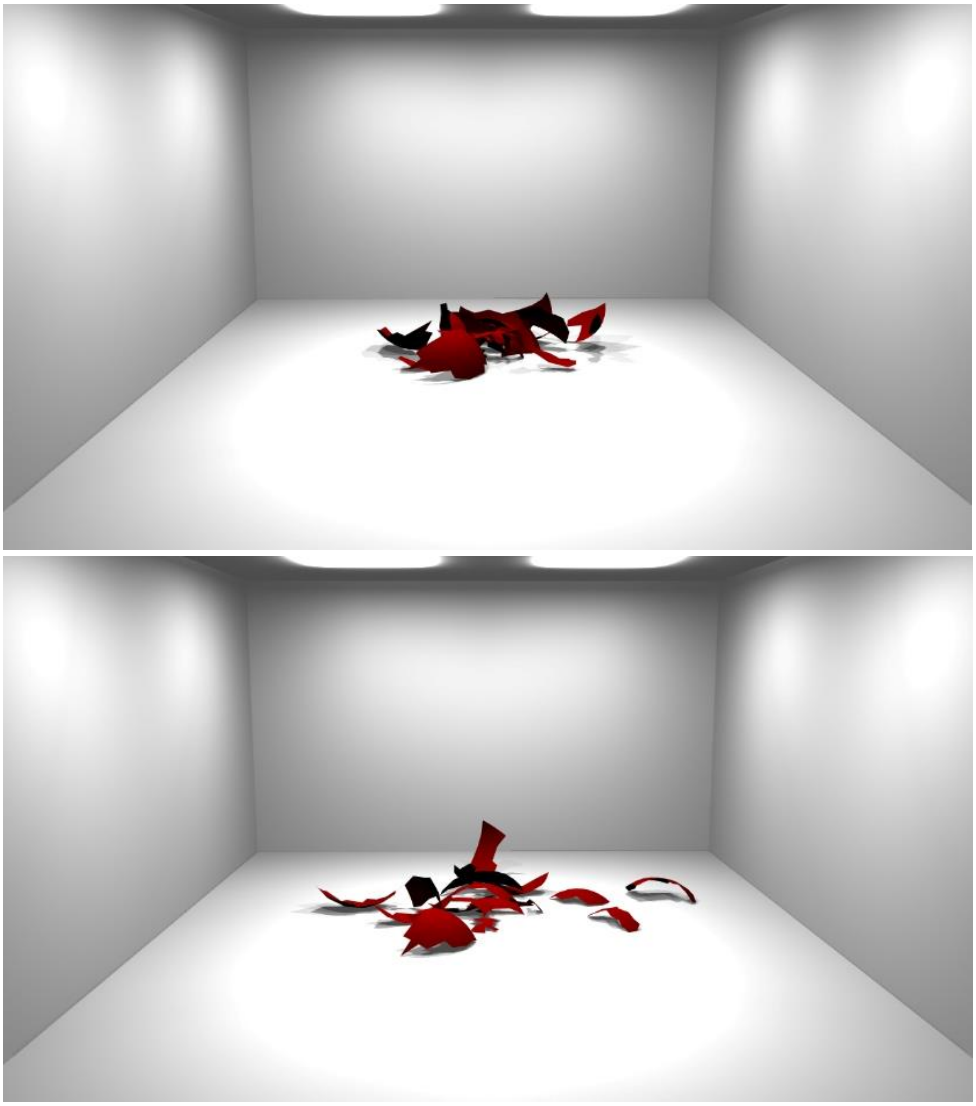
Slika 6.6. *Lom punog tijela prikazan na kugli (u Mayi i u render prozoru). Lomne su linije ciljano kreirane jednako kao kod primjera loma površine.*



Slika 6.7. *Napuknuće prikazano na ravnini (u Mayi i u render prozoru).*

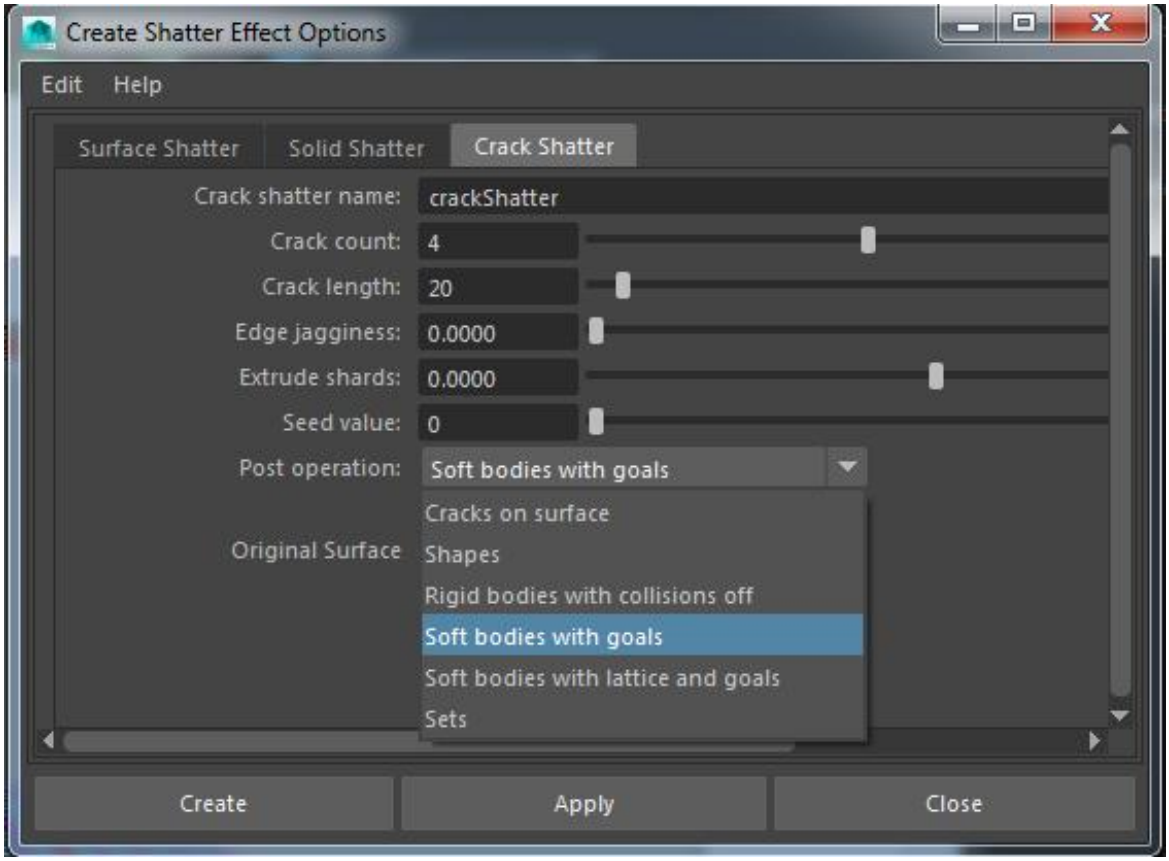
U videoanimaciji bez većih problema napravljen je primjer lomljenja korištenjem loma površine (slika 6.8.), iako je realističnost njegovih fragmenata upitna. Važno je napomenuti da se prije kreiranja bilo kojeg ugrađenog efekta loma obavezno mora obrisati povijest objekta – bez da se to učini, efekt ne funkcionira.

Lom punog tijela nikako nije mogao biti uvršten u videoanimaciju s obzirom da se program rušio pri animiranju loma kugle na samo tri fragmenta. Moguće je bilo jedino napraviti animaciju u kojoj vidno već razbijena kugla pada na pod, ali i renderiranje takvog videa trajalo bi iznimno dugo. Također, problem je bio što je predmet ovako razbijen često prolazio kroz pod prostorije: ako ne bi u potpunosti prošao kroz njega, uronio bi dio. Takvo ponašanje nije bilo objašnjivo s obzirom da je pod definiran kao kruto tijelo i pravilno mu je postavljeno polje kolizije. Isto tako, druga tijela kroz njega nisu prolazila dok lomljeno puno tijelo jest. S obzirom na sve navedeno, nije bilo moguće pripremiti primjer koji se može prikazati u videoanimaciji.



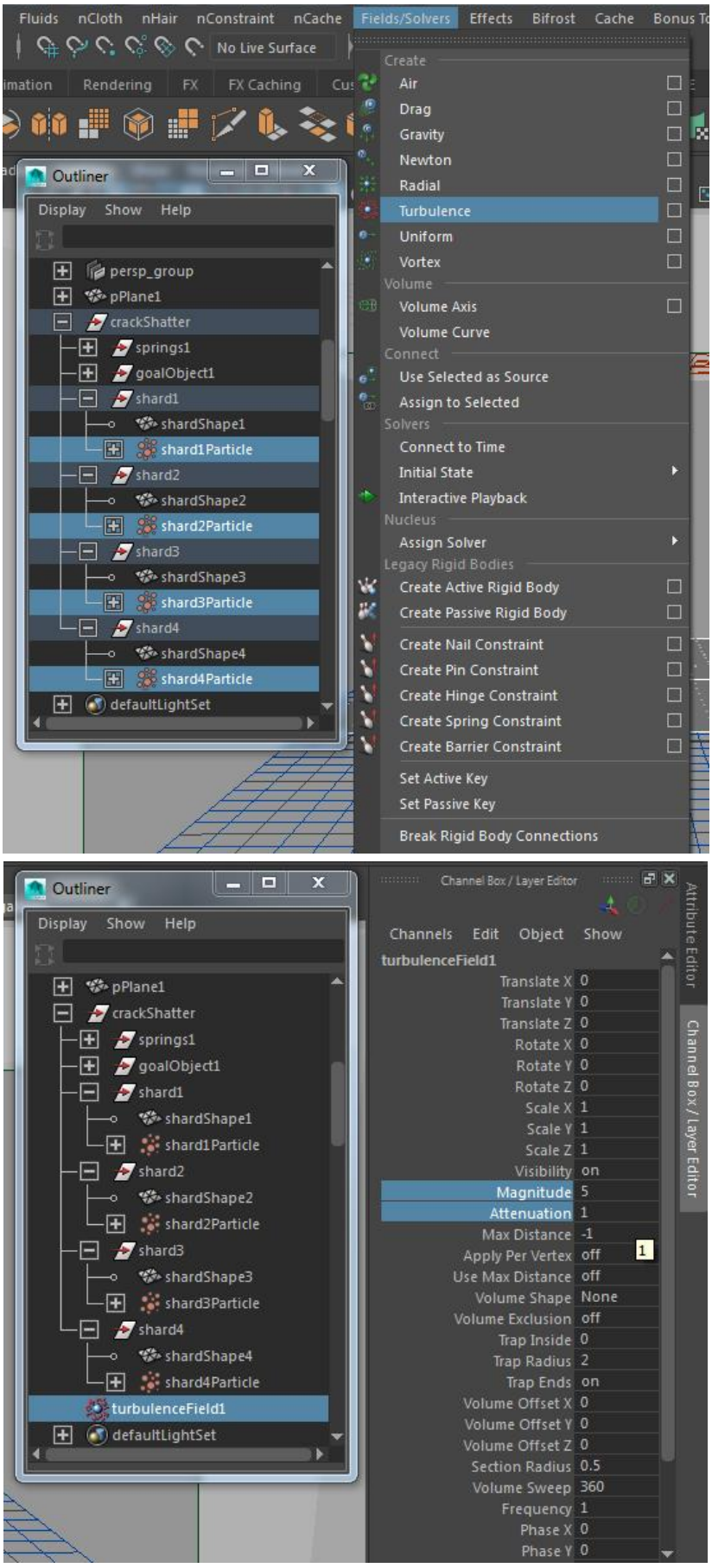
Slika 6.8. *Lom površine demonstriran na kugli. Kadrovi iz videoanimacije.*

Posljednji tip loma ugrađenog alata jest napuknuće. Njegovo kreiranje nije zahtjevno, a lom može biti relativno kvalitetan za kreiranje loma npr. gipsane ploče. Ako se primjene razne sile, moguće je kreirati i efekte nalik pucanju i deformiranju aluminija, papira ili nekih sličnih lomljivih i deformabilnih materijala. Jedan takav deformabilni lom napravljen je i za potrebe izrade videoanimacije. Na ravnini je kreiran efekt napuknuća čiji fragmenti su meka tijela s ciljnim objektima (slika 6.9). Odabir ostalih karakteristika kombiniran je dok izgled loma nije bio zadovoljavajući.



Slika 6.9. *Odabir željene opcije u polju post operation.*

Odabirom opcije prikazane na slici 6.9. Maya je za objekt automatski kreirala i nParticles. Njima su potom dodijeljena objektna dinamička polja turbulencije uz podešavanje njihove magnitude na 5 i prigušenosti na 1. Taj proces prikazan je na slici 6.10. Iako izrada ovoga nije bila suviše zahtjevna, problem je nastao pri renderiranju. Iako je Maya pravilno prikazivala animaciju, niti jedan alat nije ju mogao uspješno renderirati. Uzrok tome nije utvrđen, a rezultat je da zbog toga primjer nije uvršten u završnu videoanimaciju.

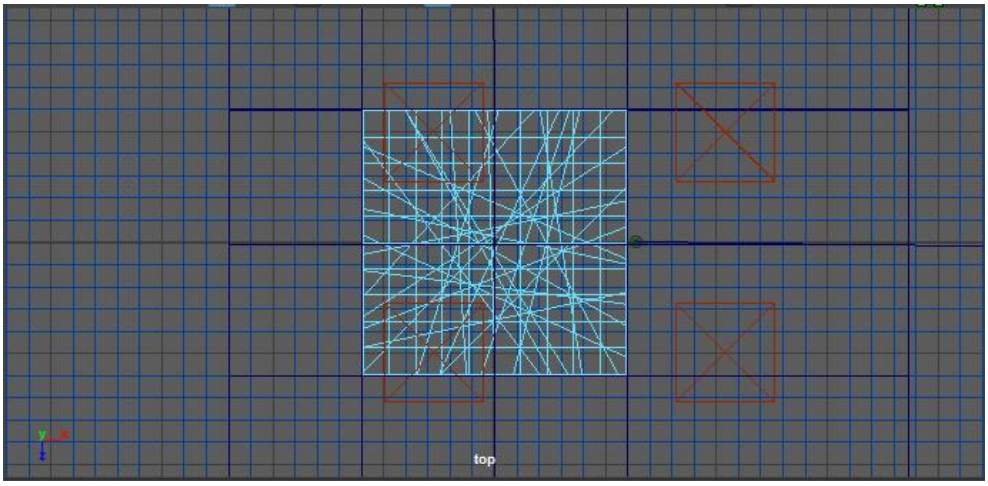


Slika 6.10. *Postavljanje objektne dinamičke sile.*

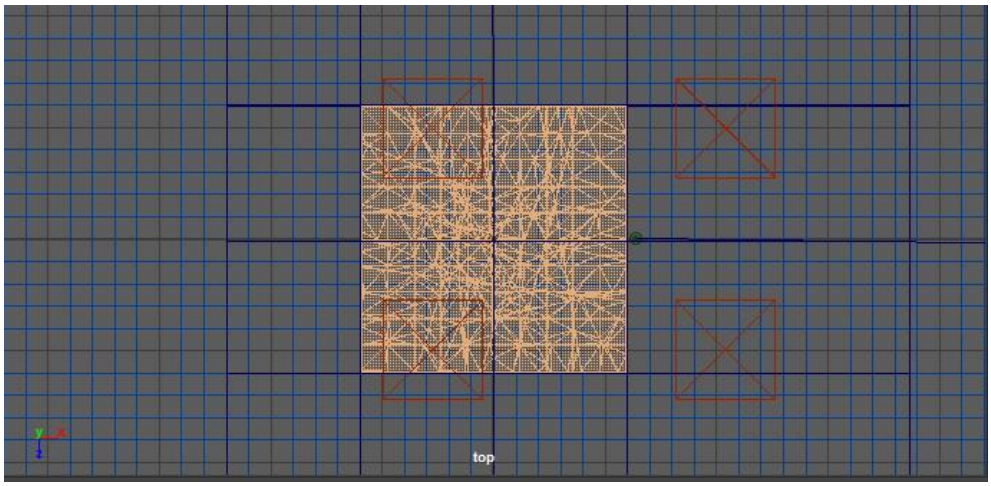
6.4. Lom pomoću nClotha

Način kreiranja loma nClothom objašnjen je u cjelini 5.3. te ga se ovdje može samo još detaljnije proširiti onime što je učinjeno pri izradi primjera za videoanimaciju. Korištena tehnika loma nClothom zapravo je tehnika koju je ranije spomenuti Mayin programer Duncan Brinsmead definirao na svome blogu. Više je raznih metoda, no Brinsmeadova je tehnika jednostavna, konkretna i široko primjenjiva te je zbog toga i odabrana. Svi napravljeni primjeri s nClothom kreirani su slično, razlikuju se samo u broju korištenih ravnina, definiranom materijalu te odabranoj lokaciji gdje lom počinje i kako se šiti, odnosno širi li se uopće.

Napravljena je ravnina i povećana je na željene dimenzije. U modulu modeliranja potrebno je odabrati *Mesh Tools* > *Multi-Cut* i izrezati ravninu na željeni broj komada. Tu fazu rada prikazuje slika 6.11. Nakon toga, bilo je potrebno triangulirati ravninu klikom na *Mesh* > *Triangulate*. Taj korak prikazan je na slici 6.12.

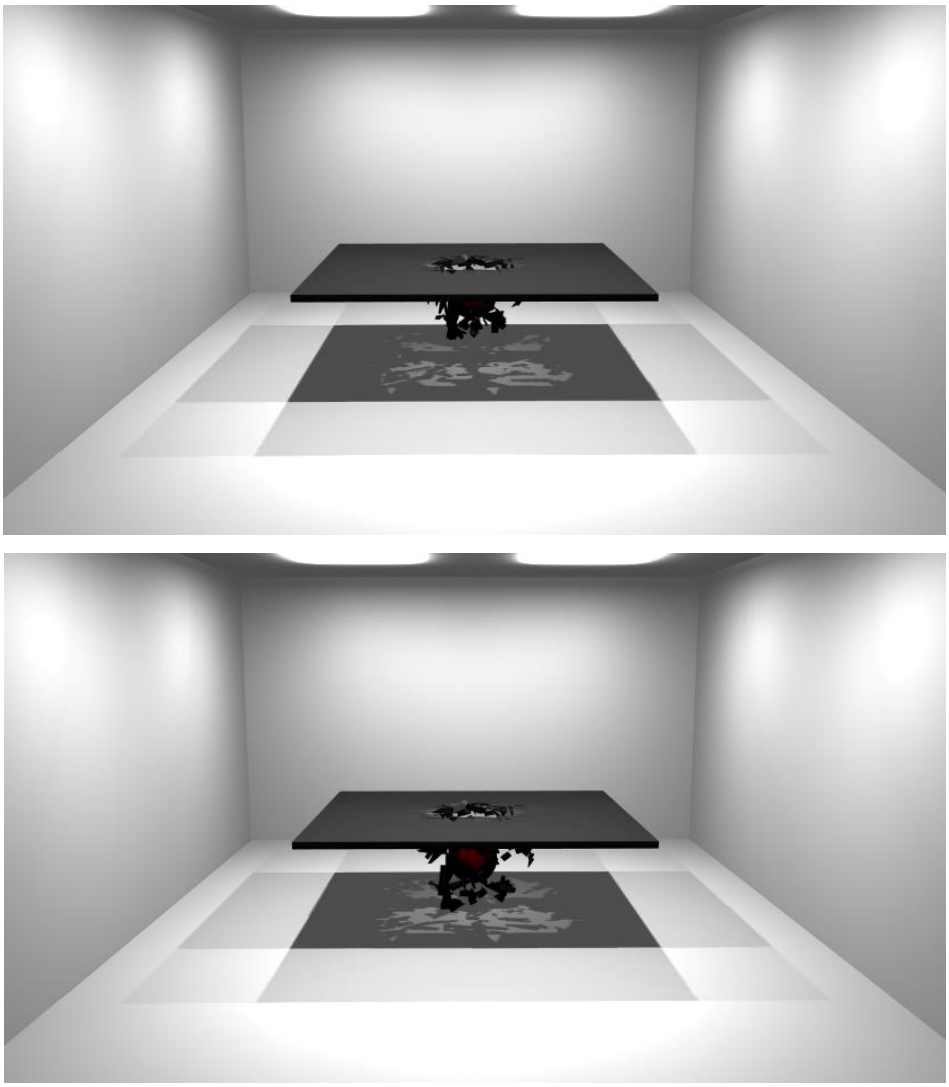


Slika 6.11. *Rezanje ravnine.*



Slika 6.12. *Triangulacija.*

Tada se odabiru samo rubovi i određuje se mjesto gdje će lom započeti, a potom se odabere *Edit Mesh* > *Detach*. Nakon toga opet se prelazi na selekciju čitavog objekta i ulazi u FX[22] modul rada te odabire *nCloth* > *Create nCloth*. U uredniku značajki (eng. *Attribute Editor*) pod novostvorenim *Nucleus* čvorom gravitaciju je potrebno promijeniti iz 9,8 u 0. Ponovno se selektira cijela ravnina i povećava joj se debljina kako bi se dobio 3D objekt. Nadalje, kreira se kugla i namješta je se iznad površine, točnije iznad mjesta na kojem je definirano da lom počinje. Dok je kugla još uvijek selektirana, odabire se *nCloth* > *Create Passive Collider*. Tada se ključnim okvirima definira translacija kugle s gornje strane površine, kroz nju i sve do poda. Naposljetku, odabiru se kugla i ravnina zajedno i klikće na *nCache* > *Create New Cache*. Rezultat ovakvog razbijanja prikazan je na slici 6.13.



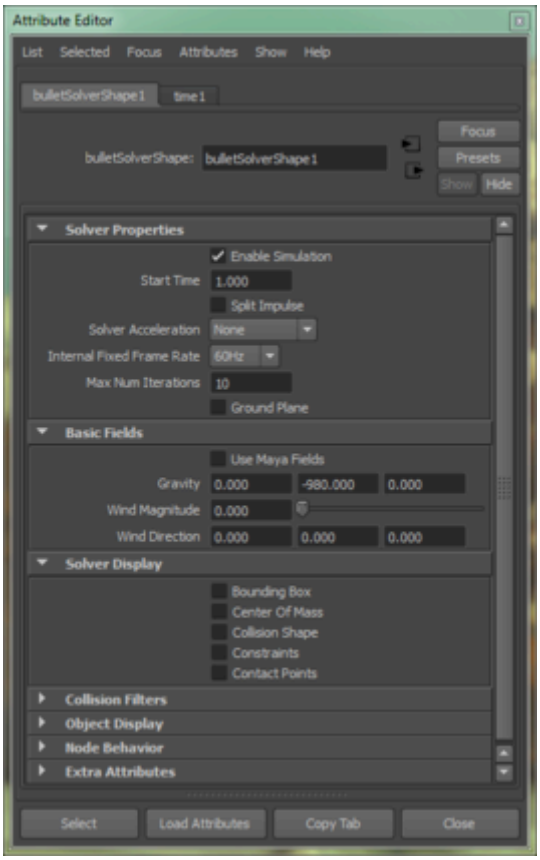
Slika 6.13. *Lom nClothom. Kadrovi iz videoanimacije.*

[22] FX modul uglavnom je ono što je u starijim iteracijama Maye bio nDynamics. S obzirom da je došlo do pregrupiranja u sučelju Maye 2016, u FX modul su uz nDynamics grupirani dodatni alati srodnog područja s ciljem okupljanja efekata na jednom mjestu.

6.5. Lom pomoću Bullet plug-ina

O radu u Bulletu već je mnogo rečeno, što u uvodu ove cjeline, što u cjelini 5.4.1. Vrlo je snažan i prilično intuitivan. Koristeći njega slomljen je kompleksniji model u radu, točnije onaj koji se nalazi i na slici u cjelini 5.4.1. Slomljen je u više od 2000 fragmenata i to koristeći mogućnost rada sa setovima krutih tijela (eng. *rigid body sets*).

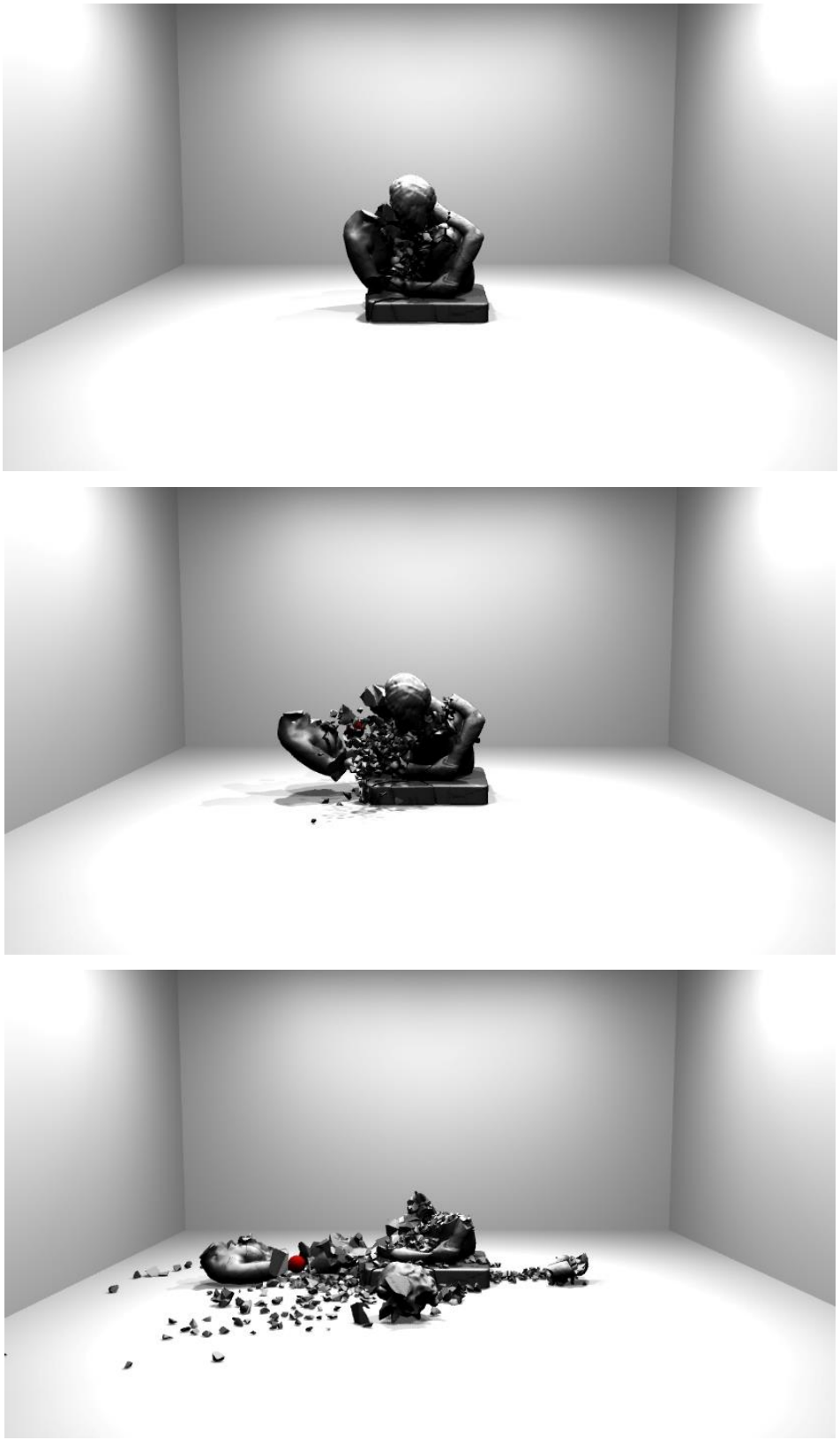
Kada je na objektu na sceni primijenjen Bulletov efekt, u uredniku značajki (eng. *Attribute Editor*) nastaje novi čvor*: bulletSolverShape*. Njegov izgled nalazi se na slici 6.14. Pomoću njega upravlja se uglavnom svim Bulletovim fizičkim postavkama, uključujući sile kao što su gravitacija i vjetar te *solver enginom* koji se koristi (OpenCL CPU, OpenCLGPU). Bitno je reći da, kako bi se vidjela svojstva setova krutih tijela, prvo se mora odabrati *solver*. Spomenuto je da Bullet ima svoja polja, no korištenjem (i) Mayinih polja postiže se veća kontrola, uz odabir opcije *Basic Fields* > *Use Maya Fields*.



Slika 6.14. *bulletSolverShape čvor u Attribute Editoru.*

Objekt se lomi prije animacije te potom lom dodatno uređuje po želji. Kada su sve postavke zadovoljavajuće, simulacijski solver za model izvozi se u obliku *alembic cache* datoteke. Nju

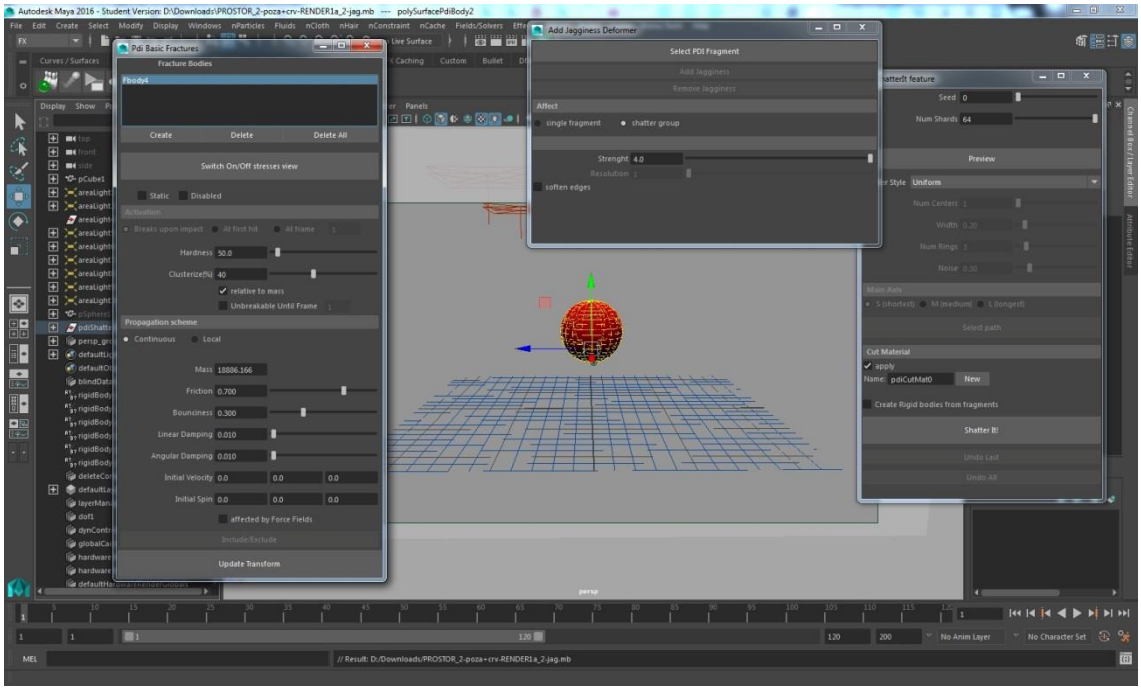
se kasnije uvozi u završenu i uređenu scenu s potpunim željenim izgledom prostora i modela. Rezultati čitavoga rada, koji je zapravo relativno dug, nalaze se na slici 6.15.



Slika 6.15. *Model razbijen Bulletom. Kadrovi iz videoanimacije.*

6.6. Lom pomoću PullDownIt plug-ina

PullDownIt plug-in opisan je u cjelini 5.4.4., a ograničenja probne verzije koju je bilo moguće isprobati navedena su u cjelini 6.1. Uzevši u obzir ta ograničenja, bilo je moguće testirati neke od ponuđenih obrazaca loma te ponašanje pri promjeni određenih karakteristika. Na slici 6.16. vidi se način kreiranja loma na kugli i ponuđene neke od postavki u PullDownItu.

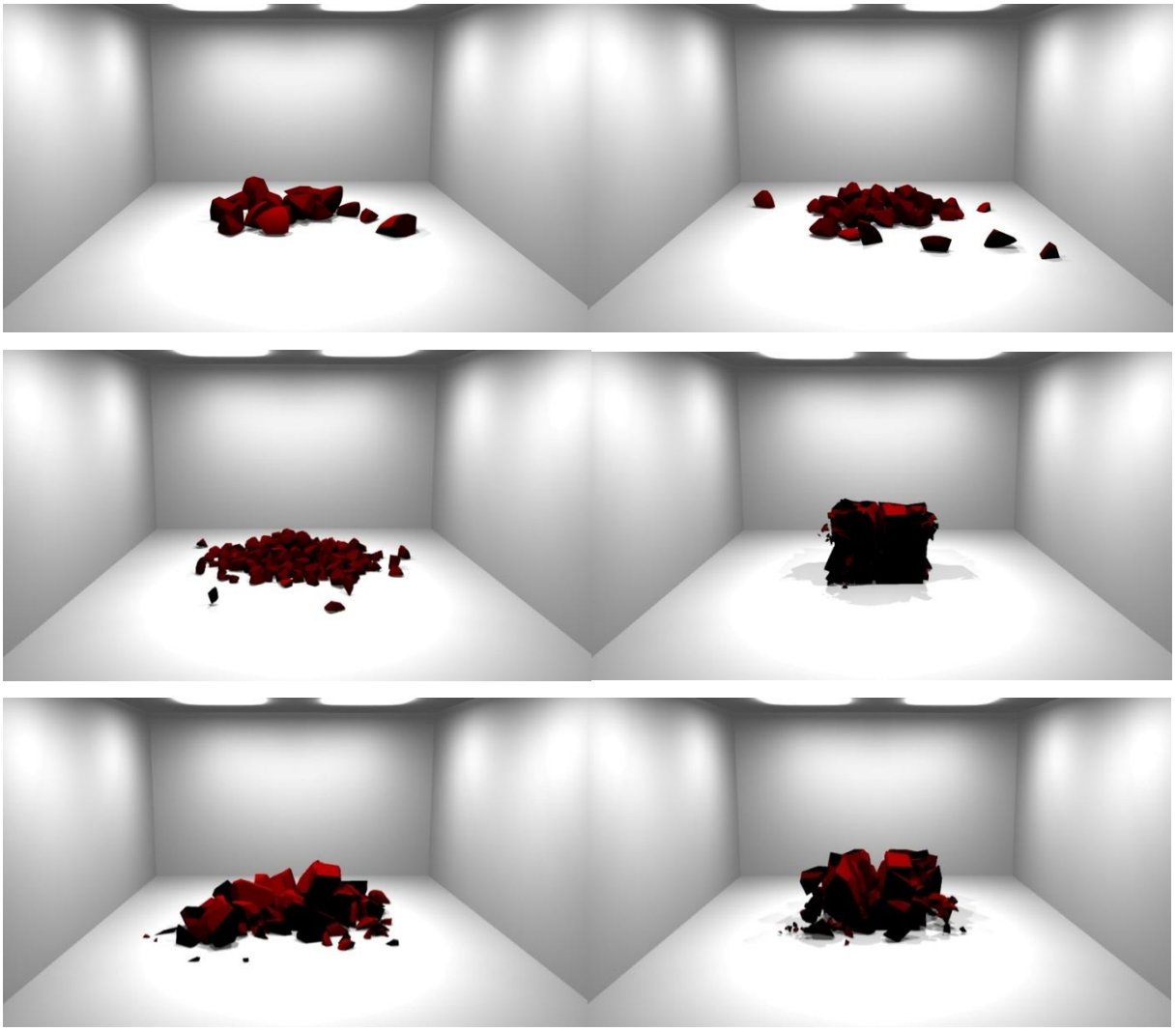


Slika 6.16. *Lomljenje objekta PullDownItom.*

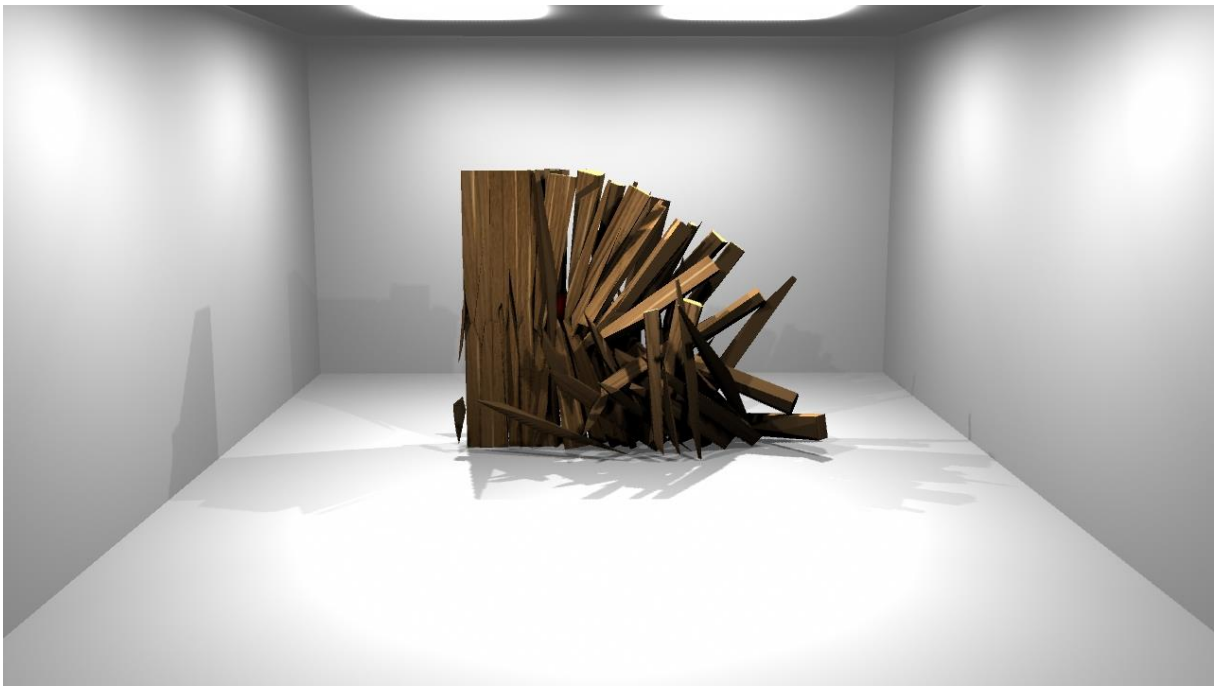
U videoanimaciji napravljeno je nekoliko primjera: kreiran je lom korištenjem različitih obrazaca loma, isprobavana je opcija skladnog ponovnog voronoi lomljenja, mijenjan je broj fragmenata i sl. U probnoj verziji nije moguće mnogo toga napraviti, no dovoljno je za prezentiranje načina rada i snage alata. Primjeri lomova iz videoanimacije nalaze se na slikama 6.17. i 6.18.

PullDownIt ima pet osnovnih prozora u kojima se kreira lom i njime upravlja. U prozoru *ShatterIt Feature* određuje se broj fragmenata, njihov raspored (isto kao *seed value* polje kod ugrađenog alata), obrazac loma i sl. U prozoru *Pdi Solver options* odabirana je opciju da Mayina mreža služi kao pod, odnosno da tijela ne prolaze kroz nju. Prozor *Manage PDI Crackers* korišten je za kreiranje tijela kojim se lomi: prvo se nacrta krivulju po kojoj će se tijelo za sudar kretati, a potom se odabire *Assign Path*, *New Cracker* i eventualno

podešavaju neke dodatne opcije. Hrapavi neravni rubovi dobivaju se tako da se selektira jedan od fragmenata i onda u prozoru *Add Jagginess Deformer* odabere polje *Add Jagginess* te dodatno namjesti željene opcije. Nakon toga, na sceni se odabere objekt koji je izlomljen na fragmente i u prozoru *Pdi Basic Fractures* odabere opcije *Create*, *Static* i *Breaks upon impact*. U tom se prozoru po potrebi namješta i druge karakteristike, kao npr. tvrdoću, klastere, distribuciju mase i sl. Ako je potrebno putanju tijela za sudar ili neravnost rubova kreirati nakon ovih postavki u *Pdi Basic Fractures* prozoru, nužno je da se zapis o tijelu u prozorčiću *Fracture Bodies* prvo obriše. Nakon željenih postavki u drugim prozorima, on se treba opet kreirati.



Slika 6.17. *Lomljenje objekata PullDownItom korištenjem obrasca voronoi lomljenja. Kadrovi iz videoanimacije.*



***Slika 6.18**. Lomljenje objekta PullDownItom korištenjem obrasca loma drveta. Kadrovi iz videoanimacije.*

U videoanimaciju pokušalo se uvrstiti razne, ali usporedive primjere kreirane svim alatima koji su bili na raspolaganju. Kreirani su različiti lomovi na istim tijelima kako bi se vidjelo koje postavke kako utječu na lom. U pisanu djelu rada uvrštene su slike nekih kadrova dijela lomova iz videoanimacije te opisane osnovne metode njihova kreiranja. Za svaki specifičan lom metode su varirale u određenim postavkama i sl., no princip rada, pa onda i njegova tehnička podloga, jednaki su. Svi primjeri renderirani su u Mayi, bez korištenja drugog softvera: iako nauštrb kvalitete slike, tako je odlučeno zbog uštede vremena, izračuni su i ovako zaista dugo trajali. Sastavljanje videa napravljeno je u programu Adobe After Effects CS6 gdje je napravljena i čitava završna obrada.

7. ZAKLJUČAK

Status i uloga Autodeskove Maye u svijetu 3D modeliranja uglavnom je neosporiva, no svoje mjesto na području dinamike tek osvaja. Kada je riječ o simulaciji dinamike mekih tijela (tkanina, kose, krzna i sl.) Maya je utabala već prilično čvrst put, no u svijetu dinamike krutih tijela još je uvijek gotovo pa promatrač. Na Mayinom dinamičkom modulu mnogo je rađeno, no značajniji je uspjeh zamijećen gotovo isključivo kada se govori o poljima i silama. Takva je situacija logična ako se u obzir uzme da je njihovo razvijanje neophodno želi li se pružiti potpuni, napredni i suvremeni alat za realistično modeliranje i animiranje mekih tijela. Pogrešno bi bilo reći da se na simulaciji krutih tijela nije radilo – točnije je reći da se na njoj *pokušalo* raditi. Iako je Autodesk zaista dugo bio ustrajan u pokušaju da razvije vlastiti RBS sustav, u iteraciji Maya 2015 dana je naznaka predaje: implementacija Bullet Physics nadogradnje, možda i najrasprostranjenijeg RBS sustava.

Koncentriramo li se isključivo na predmet rada, kôd kojim se definira alat za efekt loma Mayin je dio već više od desetljeća i gotovo da uopće nije mijenjan – a malo je vjerojatno da je tome tako jer je Autodesk njime u potpunosti zadovoljan. Bilo kakva primjenjivost toga alata upitna je: spor je do razine da se dio njegovih kreiranih mogućnosti ne može ni koristiti, a ni realističnost njegovog loma nije zadovoljavajuća. Alat bi mogao biti djelomično koristan u svrhe modeliranja, za kreiranje nasumičnih krhotina, ali i ta je primjenjivost upitna s obzirom da postoji niz besplatnih, a brzih plug-inova koji objekte lome na željene načine. Uz ove okolnosti, neizbježno je postaviti pitanje je li ovakav alat za efekt loma još uvijek Mayin dio samo zato da bi u njoj postojao ugrađeni alat kojim se lomi. Takva kružna referenca vrlo brzo dovodi do potpunog odustajanja od korištenja toga efekta, a kao alternativa uvijek se primjenjuju plug-inovi. Njihov nepregledan popis doista nudi nepresušan izvor alata primjenjivih i u najspecifičnijim zadacima. Plug-inovi testirani u ovome radu – PullDownIt i Bullet Physics – doista su primjenjivi i čine točno ono što se od njih očekuje: lome objekt onako kako im se zada i čine to kvalitetno. Oba pružaju dobru kontrolu nad lomom i nude niz finih postavki. PullDownIt je koristan za kreiranje loma za animacije zbog jednostavnog sučelja i brzog rada, a Bullet Physics može se više koristiti za dinamičke destrukcije jer dopušta i kvalitetan rad sa setovima te je vrlo snažan.

Ako je ipak potrebno objekt lomiti bez korištenja dodatnih alata, lom se može kreirati korištenjem nClotha – tu se opet svijetla s RBS-a okreću na meka tijela. nCloth jest Mayin snažan i kvalitetan modul za rad s mekim tijelima – simulator je zapravo toliko snažan i dopušta toliko kontrole da se njime mogu simulirati čak i kruta tijela, do određenog stupnja.

Lom nClothom ipak nije dovoljno realističan za tipove loma svih materijala niti je primjenjiv u svim situacijama, a i lom ovisi o ručno kreiranim lomnim linijama. Shodno tome, za kvalitetan lom korištenje plug-inova neizbježno je.

S obzirom na sve rečeno, valja zaključiti da, dok je Maya pionir kada je u pitanju simulacija i dinamika mekih tijela, nije pravi odabir za rad s krutim tijelima. Pokazatelji govore da je i Autodesk toga svjestan te stoga čelno mjesto svoga RBS modula prepušta Bulletu – dokazano kvalitetnom sustavu koji je svoje mjesto našao čak i u Houdiniji, programu koji je, u svijetu 3D modeliranja i animacije, za kruta tijela ono što je Maya za meka.

Osim što konkretnije pozicioniraju Mayu u svijetu RBS-a i 3D grafike općenito, rezultati ovoga rada daju možda i značajniji rezultat: pokazuju da niti jedan alat nije sposoban riješiti sve zadatke. Iako savršeno primjenjiv na jednom području, alat može biti u potpunosti beskoristan na drugome. Takva situacija objašnjiva je kako ubrzanim razvojem tako i problemom kompatibilnosti: kada bi program sadržavao sve alate, zauzimao bi iznimno mnogo memorije i ostalih računalnih resursa, a većina korisnika koristila bi tek 30% njegovih mogućnosti. Uz takvu situaciju, racionalno je ponuditi softver koji možda ne može sve, ali nudi osnovne alate za osnovne probleme te poznaje implementacijske metode za uvođenje sofisticiranih nadogradnji za specifične probleme. Pitanje koje ostaje jest: je li Autodesk odlučio biti racionalan i problem simulacije krutih tijela u Mayi svjesno prepustio Bulletovim developerima, ili je naprosto odustao od njegova rješavanja odlučivši profilirati Mayu kao program za simulaciju mekih tijela, gdje se RBS koristi samo kada je nužno?

Potpis studenta:_____ Datum: 9. rujna 2015.

8. Literatura

[1] Gartner. URL: www.gartner.com/ [dostupno 4. srpnja 2015.]

[2] Quake Wiki. URL: http://quake.wikia.com/ [dostupno 3. svibnja 2015.]

[3] Moby Games. URL: http://www.mobygames.com/game/windows/quake-4/screenshots/gameShotId,310210/ [dostupno 8. svibnja 2015.]

[4] Ray Wenderlich: Trigonometry for Game Programing, Tutorials for Developers & Gamers, 2013. URL: www.raywenderlich.com/35866/trigonometry-for-game-programming-part-1 [dostupno 23. ožujka 2015.]

[5] Raphael Hettich: Approaches to destruction effects in real-time computer graphic, Završni rad, Fakultät für Informatik und Wirtschaftsinformatik, Karlsruhe, 2013.

[6] Battlefield Wiki. URL: http://battlefield.wikia.com/ [dostupno 3. svibnja 2015.]

[7] Hey, A Message Bord. URL: http://denkirson.proboards.com/thread/2757?page=1#43105, [dostupno 10. svibnja 2015.]

[8] Game Crash, Battlefield 3: Back To Karkand Expansion Impressions. URL: http://gamercrash.com/2011/12/21/battlefield-3-back-to-karkand-expansion-impressions/ [dostupno 18. svibnja 2015.]

[9] Eurogamer, Tech Analysis: Battlefield 3 PC. URL: www.eurogamer.net/articles/digitalfoundry-battlefield-3-pc-tech-analysis?page=3 [dostupno 19. svibnja 2015.]

[10] Gustav Hagerling, VFX Artist. URL: http://gustav.hagerling.se/tag/screenshots [dostupno 27. svibnja 2015.]

[11] HotHardware, New Crysis 3 Trailer Unbelievably Gorgeous Thanks to CryEngine 3. URL: http://hothardware.com/news/new-crysis-3-trailer-unbelievably-gorgeous-thanks-to-cryengine-3 [dostupno 2. lipnja 2015.]

[12] TOOMBLOG: aggression in it's most elegant form, Crysis 3 Tries Really Hard to Take the First-Person Shooter to the Next Level URL: http://toomblog.com/tag/online-gaming/ [dostupno 2. lipnja 2015.]

[13] nVidia GameWorks, PhysX Destruction. URL: https://developer.nvidia.com/destruction [dostupno 10. lipnja 2015.]

[14] Mike Seymour: Art of Destruction (or Art of Blowing Crap Up), FXguide, 2011. URL: www.fxguide.com/featured/art-of-destruction-or-art-of-blowing-crap-up/ [dostupno 1. veljače 2015.]

[15] Ian Failes: 2012: Disaster Porn, FXguide, 2009. URL: www.fxguide.com/featured/2012_Disaster_Porn/ [dostupno 4. veljače 2015.]

[16] SIGGRAPH2011: Make it home, Vancouver, 2011. URL: http://www.siggraph.org/s2011/ [dostupno 21. svibnja 2015.]

[17] Sony CSL: Higher Dimensions for 3D. URL: www.sonycsl.co.jp/person/nielsen/visualcomputing/chapters/higherdimensions.html [dostupno 11. srpnja 2015.]

[18] Wikipedia: Constructive solid geometry. URL: https://en.wikipedia.org/wiki/Constructive_solid_geometry [dostupno 15. travnja 2015.]

[19] Khaled Mamou: HACD: Hierarchical Approximate Convex Decomposition, 2011. URL: http://kmamou.blogspot.com/2011/10/hacd-hierarchical-approximate-convex.html [dostupno 19. travnja 2015.]

[20] Jörg Krämer: Delaunay Triangulation In Two and Three Dimensions, Diplomski rad, Universität Tübingen, Tübingen, 1995.

[21] Ian Failes: Dark of the Moon: ILM's magic, in 3D, FXguide, 2011. URL: www.fxguide.com/featured/dark-of-the-moon-ilms-magic-in-3d/ [dostupno: 10. travnja 2015.]

[22] The Hollywood reporter. URL: www.hollywoodreporter.com/news/michael-bay-transformers-dark-of-the-moon-269204 [dostupno 21. kolovoza 2015.]

[23] Side Effects Software: Houdini. URL: www.sidefx.com/index.php?option=com_content&task=view&id=385&Itemid=190 [dostupno 28. lipnja 2015.]

[24] Mike Seymour: Harry Potter: The MAgic of Double Negative, FXguide, 2007. URL: www.fxguide.com/featured/harry_potter_the_magic_of_double_negative/ [dostupno 26. srpnja 2015.]

[25] Emanuele Goffredo: A Tool for Procedural Destruction in Houdini Shattering + Dynamics, Diplomski rad, N.C.C.A. Bournemouth University, 2010.

[26] Autodesk 3ds Max. URL: http://docs.autodesk.com/3DSMAX/16/ENU/3ds-Max-Help/index.html [dostupno: 4. kolovoza 2015.]

[27] Filippo Veniero: 3D Portfolio. URL: www.ifilgood.it/ [dostupno: 8. kolovoza 2015.]

[28] Creative Bloq: Control how objects shatter in Blender 2.6. URL: www.creativebloq.com/control-how-objects-shatter-blender-26-8134076 [dostupno: 20. srpnja 2015.]

[29] Digital Tutors: Using the Cell Fracture Feature in Blender to Create Advanced Dynamics. URL: http://blog.digitaltutors.com/using-cell-fracture-feature-blender-create-advanced-dynamics/ [dostupno: 2. svibnja 2015.]

[30] Autodesk Maya. URL: https://en.wikipedia.org/wiki/Autodesk_Maya [dostupno: 16. veljače 2015.]

[31] Digital Tutors: Introductory look at the Bullet physics engine in Maya 2016 URL: http://blog.digitaltutors.com/introductory-look-at-the-bullet-physics-engine-in-maya-2016/ [dostupno: 13. ožujka 2015.]

[32] Autodesk Maya 2016. URL: http://help.autodesk.com/view/MAYAUL/2016/ENU/ [dostupno: 20. svibnja 2015.]

[33] Azhagu Raj Pandian: Creating Glass Shattering in Maya. URL: www.dreamsin3d.com/glass_tutorial.html [dostupno: 26. svibnja 2015.]

[34] Caroline Irwin: Comparison of Shatter Effects in Autodesk Maya with nCloth and DMM Plugin, Završni rad, Faculty of Engineering and Sustainable Development, Högskolan i Gävle, 2012.

[35] Autodesk Maya: Bullet Physics Simulation Plug-in. URL: knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2015/ENU/Maya/files/GUID-88C5DD26-97D2-4972-A3DC-212FEE514175-htm.html [dostupno: 14. ožujka 2015.]

[36] Digital Tutors: Shattering a Statue Using Bullet Physics in Maya. URL: www.digitaltutors.com/tutorial/2062-Shattering-a-Statue-Using-Bullet-Physics-in-Maya [dostupno: 3. srpnja 2015.]

[37] Mike Seymour: DMM: FEA for VFX, FXguide, 2012. URL: www.fxguide.com/featured/dmm-fea-for-vfx/ [dostupno: 1. ožujka 2015.]

[38] MPC. URL: www.moving-picture.com/ [dostupno: 21. svibnja 2015.]

[39] Samuel Tissot: Wall Destruction. URL: http://samueltissot.com/blog/ [dostupno: 1. kolovoza 2015.]

[40] Fracture FX. URL: www.fracture-fx.com [dostupno: 27. ožujka 2015.]

[41] Polygons, pixels and paint: A Look at Fracture FX - Maya's Most Complete Commercially-available Destruction System. URL: http://polygonspixelsandpaint.tumblr.com/post/21327502894 [dostupno: 1. travnja 2015.]

[42] Thinkinetic. URL: www.pulldownit.com/ [dostupno: 1. travnja 2015.]

[43] Adam Guzovski: Pulldownit review. URL: www.evermotion.org/tutorials/show/7962/pulldownit-review [dostupno: 1. travnja 2015.]

[44] Lesterbanks: Thinkinetic Releases PullDownIt 2 Which Features Jagged Surface and Edge Fractures. URL: http://lesterbanks.com/2012/10/thinkinetic-releases-pulldownit-2-which-features-jagged-surface-and-edge-fractures/ [dostupno: 1. Travnja 2015.]

[45] CreativeCrash: Ninja Fracture 5.2.0. URL: www.creativecrash.com/maya/script/ninja-fracture [dostupno: 2. svibnja 2015.]

[46] Duncan's Corner. URL: http://area.autodesk.com/blogs/duncan [dostupno: 23. srpnja 2015.]

[47] FXguide, VXF forums. URL: www.fxguide.com/forums/ [dostupno: 22. rujna 2015.]