



Počítačová grafika

Simulace elastických těles pomocí vázaného časticového systému

11. prosince 2016

Autoři: Jan Kohút,
Martin Knotek,
Tomáš Juřica,

xkohut08@stud.fit.vutbr.cz
xknote11@stud.fit.vutbr.cz
xjuric22@stud.fit.vutbr.cz

Fakulta Informačních Technologií
Vysoké Učení Technické v Brně

Obsah

Zadání	2
Nejdůležitější dosažené výsledky	3
Realistická simulace pružinovým systémem	3
Detekce kolizí	3
Zvláštní použité znalosti	5
Práce na projektu	6
Rozdělení práce v týmu	6
Co bylo nejpracnější	6
Zkušenosti získané řešením projektu	6
Autoevaluace	7
Ovládání vytvořeného programu	8
Technologie potřebné pro spuštění programu	8
Programy a služby použité při tvorbě	8
Obsluha programu	8
Doporučení pro budoucí zadávání projektů	9
Literatura	10

Zadání

Cílem projektu je vytvořit maximálně realistickou simulaci elastických těles pomocí částicově vázaného systému. Součástí simulace je taktéž detekce kolizí objektů a reakce na ně. Simulaci bude možné spustit a sledovat v reálném čase.

- Realistická simulace elastických těles pomocí částicového systému
 - Těleso se skládá z hmotných bodů. Tyto body na sebe vzájemně působí. Působící síly jsou simulovány pružinami.
 - Vnitřní opora tělesa je také simulována tlakem ideálního plynu o objemu tělesa.
- Detekce kolizí objektů a následná reakce
 - Kolize detekovány v reálném čase
 - Realistické reakce objektů po kolizi včetně vzájemného tření
- Interaktivní grafické zpracování
 - Realistické osvětlení, stíny
 - Možnost zábavnou formou přidávat objekty a způsobovat tak kolize.
- Možnost načítání externích modelů.

Nejdůležitější dosažené výsledky

Realistická simulace pružinovým systémem

Abychom simulovali pružnost, přistupujeme k tělesu jako k množině hmotných bodů rovnoměrně rozmístěných po plášti tělesa. Tyto body mají mezi sebou vazby, které připodobňujeme pružinám a modelujeme je se stejnými fyzikálními vlastnostmi. Třetí složkou simulace pružnosti je vnitřní tlak v tělese. Ten modelujeme jako ideální plyn podporující tvar tělesa zevnitř.

V kontextu počítačové grafiky je těleso reprezentováno body (vertices), z těchto bodů jsou složeny jednotlivé trojúhelníky. Z jakých bodů se daný trojúhelník skládá je uloženo ve indices. Celkový objekt se skládá z konečné množiny těchto trojúhelníků.

Toho využíváme v naší simulaci, Vertices a hmotné body jsou shodné. Jeden hmotný bod je pak ve vztahu (spojen pružinou) se všemi ostatními body. Ačkoliv se jedná o výpočetně nejnáročnější volbu, bylo při ní dosaženo nejrealističtějších výsledků deformace.

V každém simulačním kroku jsou pro každý hmotný bod vypočítány následující síly:

1. Gravitační síla
2. Síla pružin a tlumičů
3. Síla vnitřního tlaku ideálního plynu

Celková síla působící na bod je součtem těchto sil. Výpočet podrobněji popsán v [4, 2]

Detekce kolizí

Pro zajištění plynulejšího chodu celé simulace jsou nejprve objekty testovány podle ohraničení AABB (Axis Aligned Bounding Box). Každý jeden objekt v simulaci má určeno své ohraničení AABB, které je použito pro hrubý odhad průniků dvou objektů.

Pokud jsou dva objekty podle AABB v kolizi, dojde na přesnější vypočtení odrazu a deformace, které probíhá po jednotlivých vertexech. Z toho vyplývá i důvod použití AABB - v případě použití bez AABB, by se všechny objekty testovaly vůči všem objektům vertex po

Síla pružiny	20.0f
Váha bodu	1.1f
Síla tlumiče	10.0f

Tabulka 1: Experimentálně nalezená konfigurace pružinového systému.

vertexu, což by vedlo k mizernému výkonu, protože by se přepočítávaly i vertexy objektů, které spolu vůbec nemusí být v kolizi.

Detailnější výpočet deformace probíhá, jak bylo uvedeno výše, vertex po vertexu a obecně v simulaci mohou nastat 2 případy - buď koliduje objekt z rovinou nebo objekt s objektem.

V případě kolize objekt-rovina dojde ke zpomalení vrcholů dotýkajících se roviny a aplikaci síly na tyto vrcholy. Výsledná síla, která tlačí vrcholy pryč od roviny, se potom šíří celým tělesem a způsobuje realistickou deformaci a pohyb objektu (např. následný odraz).

V případě kolize objekt-objekt dochází k výpočtu odražení jednotlivých vrcholů naivním přístupem - tedy vrchol po vrcholu. Každý z vrcholů jednoho objektu je testován vůči kolizi s každým vrcholem druhého objektu. Pokud jsou některé dva vrcholy v kolizi, dojde k výpočtu příslušného vektoru rychlosti, výpočtu síly a její aplikaci na dané vrcholy, a také ke korekci pozic obou vrcholů.

Aplikací síly na jeden vrchol způsobí, díky spojení všech vrcholů pružinkami, její šíření dál celým tělesem a tedy i věrohodnou deformaci a pohyb celého objektu.

Ke korekci pozice obou vrcholů musí dojít z toho důvodu, aby se předešlo pohybování vrcholů "sem a tam".

Zvláštní použité znalosti

K řešení projektu byly využity poznatky ze středoškolské fyziky a to newtonovy zákony, vlastnosti pružin a plynů.

Práce na projektu

Rozdělení práce v týmu

- **Jan Kohút:** Osvětlení scény.
- **Martin Knotek:** Logická struktura simulace, detekce kolizí objektů, fyzikální stránka simulace.
- **Tomáš Juřica:** Otevírání a tvorba .obj souborů fyzikální stránka simulace.

Co bylo nejpracnější

- **Jan Kohút:** Správné načtení normál a vrcholů z OBJ souboru do bufferů tak, aby se objekty správně osvětlily.
- **Martin Knotek:** Detekce kolizí objektů a realistická reakce.
- **Tomáš Juřica:** Jak reprezentovat objekt pevnými body a pružinami pro jeho realistickou deformaci.

Zkušenosti získané řešením projektu

Řešením projektu jsme se prakticky naučili základy OpenGL a počítačové grafiky (práce v souřadnicovém systému, aplikace transformací, práce s kamerou, tvoření shaderů, implementace ambientního, difúzního a spekulárního osvětlení). Dále jsme si zopakovali fyzikální vlastnosti pružin a plynů. Za zmínku také stojí zdokonalení se v týmové spolupráci, v diskuzi různých přístupů řešení a lepší ovládnutí githubu,

Autoevaluace

Technický návrh (92%): Počátečnímu návrhu a studiu problematiky byl věnován velký podíl času, což se při další práci mnohonásobně vrátilo. Z pohledu architektury byly při vývoji nutné jen minimální zásahy.

Programování (85%): Kód je členěn do jednotlivých tříd, členové týmu se zdokonalili v psaní kvalitního C++ kódu.

Vzhled vytvořeného řešení (60%): Osvětlené objekty nevrhají stíny, osvětlení nefunguje správně u určitého typu objektů z .obj souboru. Scéna je složena ze základních tvarů, což je však k demonstraci problému dostačující.

Využití zdrojů (83%): Byla nastudována literatura k dostatečnému pochopení a implementaci problematiky. Při dalším studiu by s největší pravděpodobností objeveno optimálnější řešení.

Hospodaření s časem (95%): Projekt byl díky Martinovi řešen už od jeho zadání a jeho řešení tak bylo rovnoměrně rozloženo.

Spolupráce v týmu (90%): Všichni členové bez problémů komunikovali.

Celkový dojem (100%): Projekt byl hodnocen jako jeden z nejzajímavějších za studium na VUT FIT. Náročnost a pracnost byla zcela vyhovující.

Ovládání vytvořeného programu

Technologie potřebné pro spuštění programu

- Projekt je napsán v jazyce C++, přeložitelný ve Visual Studiu 2015.
- Ke spuštění je potřeba mít staženy a správně nalinkovány knihovny SDL2, glew, opengl. Tyto knihovny, společně s kompletním řešením i .sln souborem jsou dostupny na <https://github.com/kohuthonza/ElasticObjects>.

Programy a služby použité při tvorbě

- Vyvíjeno v prostředí Visual Studio 2015.
- K týmové spolupráci při vývoji byl využit GitHub.
- Tým nejčastěji komunikoval pomocí Facebook Messengeru.

Obsluha programu

Po spuštění se uživatel nachází před předdefinovanou scénou s množstvím objektů, které mezi sebou po určitém čase začnou kolidovat.

Pohyb po scéně: Změna úhlu kamery je možná pohybem myši. Polohu kamery lze měnit šipkami.

Přidání nových objektů: Přidání nových objektů je možné mezerníkem. Po stisknutí je z pozice kamery směrem do scény vyslán objekt koule.

Doporučení pro budoucí zadávání projektů

- Při zadávání projektů jsme ocenili volné a velice atraktivní zadání a také jejich pestrost.
- V průběhu projektu vyvstaly nejasnosti ohledně datumu odevzdání, možnosti pozdního odevzdání, oficiální zdroje byly často nekonzistentní a neaktualizované.
- Oceňujeme také účinnou konzultaci, která nám během 15 minut dala velice použitelný přehled daného tématu.

Literatura

- [1] GameDev.net. Nehe opengl tutorials.
- [2] Maciej Matyka. How To Implement a Pressure Soft Body Model. .
- [3] Maciej Matyka and Mark Ollila. Pressure model of soft body simulation. In *The Annual SIGRAD Conference. Special Theme-Real-Time Simulations. Conference Proceedings from SIGRAD2003*, number 010, pages 29–33. Linköping University Electronic Press, 2003.
- [4] Jaruwan Mesit, Ratan Guha, and Shafaq Chaudhry. 3d soft body simulation using mass-spring system with internal pressure force and simplified implicit integration, 2007.
- [5] Wikipedia. Elastic collision, 2016. [Online; Navštíveno 11.12.2016].