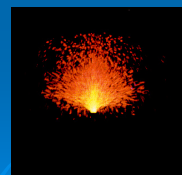


Částicové systémy

I.Kolingerová

1. Částicové systémy – základní popis
2. Generování částic
3. Atributy částic
4. Dynamika částic
5. Rušení částic
6. Renderování částic
7. Systémy se vzájemnou interakcí částic
8. Příklady



KPG2/32

Literatura:

- W.T.Reeves: Particle Systems – a Technique for Modeling a Class of Fuzzy Objects, ACM Transactions on Graphics, Vol. 2, No. 2, 1983, pp.91-108
- T.Jelen: Modelování pohybujících se geometrických objektů pomocí částicových systémů, bakalářská práce, ZČU Plzeň, 2003

1. Částicové systémy – základní popis

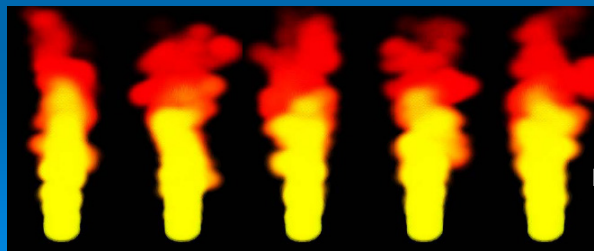
- Metoda pro generování fuzzy objektů typu oheň, ohňostroj, exploze, tráva, mraky, voda ...
- **Fuzzy objekty** – obtížné, nemají hladké, dobře definované povrchy, pohyb nelze popsat afinními transformacemi
- **Částicový systém** - objekt modelován jako mrak primitiv. částic definujících objem objektu
- Částicím lze přiřadit libovol. dynamiku, libovolný počítačový model vzhledu
- Generující tvar: koule, kružnice, obdélník,...

Ukázka: Particles.avi, Particle2.avi

- **Odlišnosti** od obvyklé reprezentace:
 - Místo množiny jednoduchých povrch. prvků mraky částic definujících objem
 - Částice se rodí a umírají
 - Není zcela deterministické, užití stochastických procesů pro vytváření a změnu vzhledu
- **Výhody:**
 - Částice jednodušší než polygon
 - Procedurální definice, řízení náhod. čísl => rychlý návrh
 - Lze vytvořit level-of-detail (LOD) podle pohledových parametrů
 - "živý" systém – dynamika snazší než s povrch. popisem

➤ **Výpočet 1 snímku:**

- Generace nových částic do systému
- Přirazení individuál. atributů novým částicím
- Likvidace částic překročivších předepsanou dobu života
- Posouvání a transformace zbylých částic podle jejich atributů
- Renderování žijících částic do frame bufferu



Hoření

➤ **Ukázka:**

bin/psys.exe

2. Generování částic

- Řízený stochastický proces určuje počet částic vstupujících v každém snímku do systému

$$N_{parts_f} = MeanParts_f + Rand() \times VarParts_f$$

aktuální počet částic ve framu f
 střed. hodnota počtu
 náhod.č. s rovnoměr. rozlož. z <0,1>
 odchylka

Nebo:

$$N_{parts_f} = (MeanParts_{saf} + Rand() \times VarParts_{saf}) \times ScreenArea$$

plocha obrazovky zakrytá objektem
 střed. hodnota počtu/plocha obrazovky
 odchylka počtu/plocha obrazovky

Počet částic lze měnit v čase:

$$Meanparts_f = InitialMeanParts_f + DeltaMeanParts \times (f-f_0)$$

současný snímek
 1.snímek, kdy systém naživu
 střed. hodnota počtu částic v 1. snímku
 rychlost změny

- podobně pro $MeanParts_{saf}$
- $VarParts$ beze změny
- Možná také změna kvadratická, kubická, stochastická ...

Hierarchie částic

- Střed. hodnota a odchylka se užijí pro skupinu potomků dané částice

3. Atributy částic

- **Počáteční pozice** – uvnitř generujícího tvaru, 3D bod + 2 úhly pro orientaci
- **Počáteční rychlost** – velikost i směr
 $\text{InitialSpeed} = \text{MeanSpeed} + \text{Rand}() \times \text{VarSpeed}$
- **Počáteční velikost** – dostane průměr a max. odchylku
- **Počáteční barva** – dostane průměr. R,G,B a max. odchylku
- **Počáteční průsvitnost** – dostane průměr a max. odchylku
- **Tvar** – koule, obdélník, ...
- **Doba života** – kladné číslo, dekrementováno

4. Dynamika částic

- Nejjednodušší: poloha funkcí času
- Složitější chování – do systému vnější síly, působící na částice
- Pohyb. rovnice:

$$v = v_0 + \int a dt$$

$$p = p_0 + \int v dt$$

p – poloha částice
 v – vektor rychlosti
 p_0, v_0 – počát. poloha a rychlost
 $a = f/m$ (vnější síla/hmotnost částice)

- Aproximace:

$$v_{n+1} = v_n + a \Delta t$$

$$p_{n+1} = p_n + v_{n+1} \Delta t$$

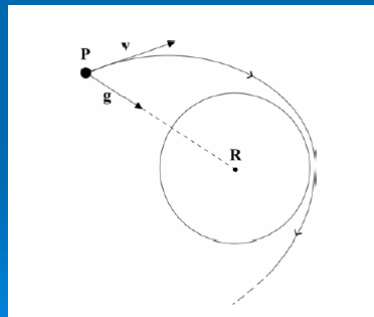
Δt – časový krok simulace

Nejčastější silou je **gravitace**:

$$f_g = mgd$$

g – gravitační zrychlení
 d – směrový vektor

- vizuální výsledek: parabolický pohyb



Pohyb částice
 kolem středu gravitace

Odpor prostředí: proti směru pohybu

$$\mathbf{f}_r = -k_r \mathbf{v}$$

k_r – konstanta odporu prostředí
 \mathbf{v} – původní vektor rychlosti

Odraz od geom. objektů:

$$\mathbf{v}' = \mathbf{v} - 2(\mathbf{v}\mathbf{n})\mathbf{n}$$

\mathbf{v}' – vektor rychlosti částic po odrazu
 \mathbf{v} – původní vektor rychlosti
 \mathbf{n} – normál. vektor odrazové plochy

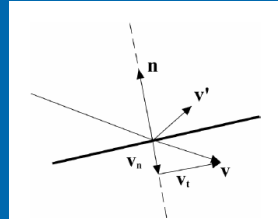
- Odraz částice až z polohy pod povrchem, do kterého pronikla během 1. kroku simulace – při větším simulačním kroku nepřírozené
- Možné řešení: dopočítání přesnější polohy odrazu z předchozí polohy částice, z toho výpočet nové rychlosti

Další zpřesnění: **elastická srážka**:

- Rychlost po odrazu rozdělena na normál. a tečnou složku

$$v_n = (vn)n$$

$$v_t = v - v_n$$



- Nová rychlost po srážce je

$$v' = (1 - \mu)v_t - \varepsilon v_n$$

μ – zmenšuje tečnou složku
– koef. tření

ε – ovlivňuje normál. složku
– koef. pružnosti

5. Rušení částic

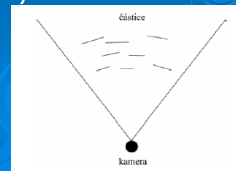
- Při dosažení doby života
- V případě, že se částice v daném směru pohne o více než o daný rozdíl



Vodopád

6. Renderování částic

- Vzájemné zakrytí částic i částic a klasických objektů, průhlednost, neviditelnost
- Možná zjednodušení:
 - část. systém rozdělit na více, aby žádný průnik s povrchově model. primitivy, složení až dodatečně
 - částice – bodové světelné zdroje – žádné stíny ani neviditelnost (pouze se přidává světlo pixelům), pouze ořezání hodnoty na max., není nutný sort podle hloubky
- Zjednodušení OK pro exploze a oheň, nevyhovující pro mraky a vodu
- Často místo částic zobrazovány texturní obdélníky (sprites)



7. Systémy se vzájemnou interakcí částic

- Vzájemné přitahování a odpuzování, detekce kolizí, pružiny, místo částic zvířata, ptáci, lidé ...
- Základní myšlenka: relativně jednoduchá pravidla chování jedince v rámci hejna:
 - Vyhýbání se kolizím
 - Přizpůsobení se pohybu jedinců v nejbližším okolí
 - Držení se v blízkosti hejna – směřování do centra blízkých jedinců
- Pro i-tou částici:



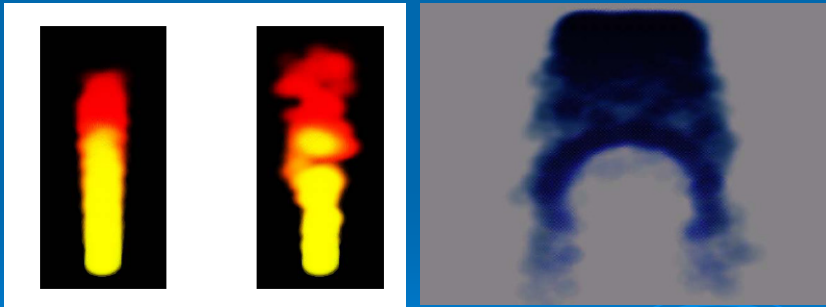
$$\mathbf{v}_i' = \mathbf{v}_i + \mathbf{a}_i \Delta t, \mathbf{a}_i = \mathbf{f}_{\text{ext}}(\mathbf{p}_i, \mathbf{v}_i) / m, \mathbf{f}_{\text{ext}} = \mathbf{f}_g + \mathbf{f}_r + \mathbf{f}_{\text{env}},$$

$$\mathbf{f}_g = m_i \mathbf{g} \mathbf{d} - \text{gravitace}, \mathbf{f}_r = -\varepsilon \mathbf{v}_i - \text{odpor prostředí},$$

$$\mathbf{f}_{\text{env}} = \mathbf{f}_{\text{env}}(\mathbf{p}_i, \mathbf{v}_i) - \text{okolí (def. uživatel)}$$

➤ Okolí

- např. Perlinův šum – způsobí nepravidelné vlnění plamene a shluk částic do obláčků => nápodoba turbulence plynu
- nebo odpudivá síla na povrchu plynu => obtékání



Plamen bez a se šumovou funkcí

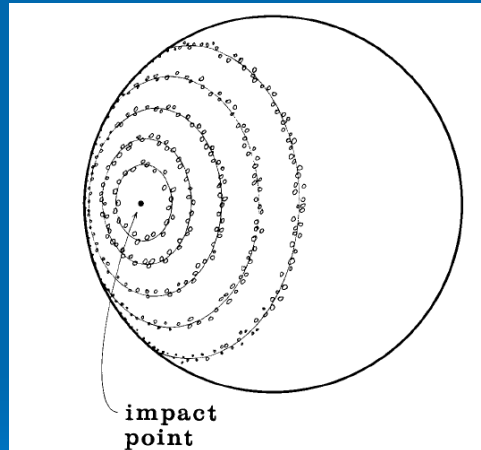
Obtékání koule částicemi

8. Příklady

Stěna ohně a exploze (1):

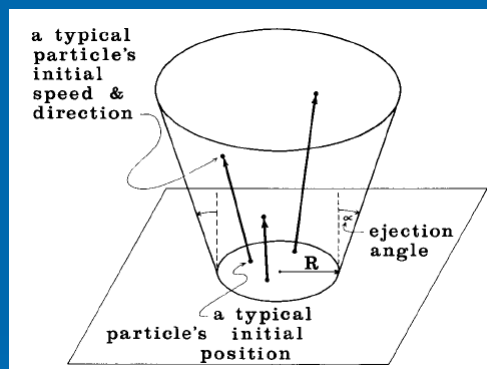
- Genesis Demo ze StarTrek II: The Wrath of Khan (Paramount, 1992) – sekvence generována v Lucasfilmu – přeměna mrtvé planety v živou výbuchem Genesis bomby – po explozi se šíří stěny ohně od bodu dopadu, jejich vlivem vznik hor atd.
- 2 úrovnňový částic. systém se středem v bodu dopadu bomby, soustředné kruhy 2. úrovní

Stěna ohně a exploze (2):



Distribuce částicových systémů na povrchu planety

Stěna ohně a exploze (3):



Ukázka částicového systému 2.úrovně (vzhled napodobuje explozi)

Stěna ohně a exploze (4):



Ukázka počáteční exploze

Stěna ohně a exploze (5):



Postupující ohňová stěna

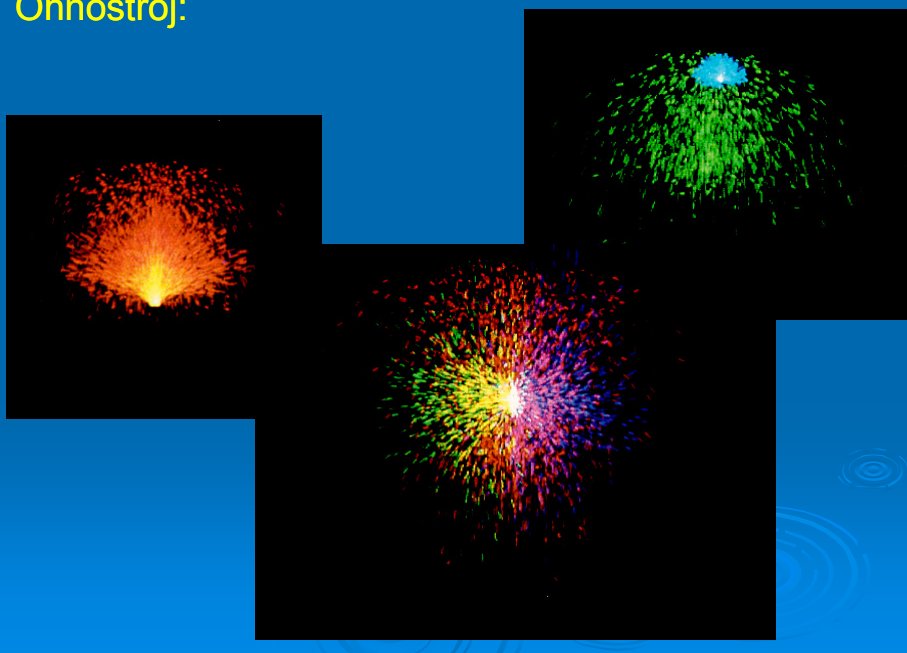
Stěna ohně a exploze (6):



Další ukázky z téže aplikace

Ukázka: Genesis

Ohňostroj:



Tráva:



3D Max: sprej, snow, superspray, vánice (blizzard), pohyb lze dále ovlivňovat pomocí tzv. space-warpů, lze přiřadit materiály, měnit je podle věku částice



2 částicové systémy - oheň a kouř (intenzita se přičítá k pozadí nebo odčítá)



Vodopád - částice se odrážejí od objektu



Sníh: uživatelem def. objekty místo částic

Ukázka: snow.m1v, water.m1v, fire.m1v

Les: plochu rozdělit na čtverce, v každém vybrat střed a náhodně posunout tak, aby zůstal v daném čtverci, cca 10% bodů se vypustí, ve zbylých se vygeneruje jednoduchý model stromu z několika válců se vhodnou texturou, výsledek z dostatečné vzdálenosti realistický

Dynamická simulace: koš míčů rozsypaný po schodech, emitující objekt – koš, částice – míče. Vlastnosti míče – počáteční rychlost, směr, pružnost, barva (textura), hmotnost, všechny pohyb směrem dolů. Míče se mohou srážet, odráží se od zábradlí atd.

Ukázka: Dynamic.avi, Dynamic2.avi

Bublinky v sodovce: generuje je vnitřní povrch sklenice, mají různou velikost, směr vzhůru, na hladině zaniknou

Užití též pro složité fyzikální simulace – **trasování částic:** pro zobrazení složitého fyz. pole bez příliš velké přesnosti, na výpočet málo času, zobrazuje se dráha částic, červeně místa, kde např. velká změna dráhy, a tedy i velké tření

Simulace růstu rostlin: vyhýbají se překážkám, reakce na světlo.