
Detekce kolizí v 3D

**© 2001-2003 Josef Pelikán
KSVI MFF UK Praha**

e-mail: Josef.Pelikan@mff.cuni.cz

WWW: <http://cgg.ms.mff.cuni.cz/~pepca/>

Aplikace CD

- **mobilní robotika**
 - plánování cesty “robota” bez kontaktu s okolím
- **animační a simulační systémy, hry**
 - test konzistence scény (v některých apl. až 1 kHz !)
 - interaktivní fyzikální modelování („akce - reakce”)
- **CAD, strojírenský průmysl**
 - robustní a numericky stabilní implementace!
- **molekulární modelování**

Úlohy

- jednoduchá **detekce kolize** mezi některými z **N** těles v 3D scéně
 - tělesa m_i a n_i mají neprázdný průnik
- detekce kolize + „**úroveň vnoření**“
 - jak hluboko do sebe tělesa pronikla (např. pro pozdější výpočet reakční síly)
- nalezení **prvního kontaktu** (čas i místo)
 - přesný výpočet časového okamžiku, ve kterém se tělesa poprvé dotkla (možnost přesnější reakce)

Podmínky

- velké množství **pohybujících** se těles
 - naopak mohou být některé části scény statické
- tělesa jsou často v **těsném kontaktu**
 - „close proximity“, problém pro většinu běžných alg.
- testy je potřeba **velmi často opakovat**
 - pro některé fyzikální simulace je někdy potřeba počítat CD i několikrát rychleji, než běží zobrazení!
 - využití koherence scény v čase (spojitost pohybu)

Data (3D scéna)

- **trojúhelníkové sítě**
 - bez jakékoli topologie („triangle soup”)
 - topologická reprezentace („triangle mesh”)
 - hierarchie LoD („Level of Detail”)
- **přesné matematické vyjádření křivek a ploch**
 - parametrické křivky/plochy (nejčastěji NURBS)
- **geometrická tělesa**
 - konečný počet typů (geometrických primitiv)
 - kombinace např. množinovými operacemi (CSG)

Hlavní přístupy

- **dělení prostoru**
 - pravidelné mřížky, oktantové, BSP, k-d stromy, ..
- **hierarchie obalových těles (BV)**
 - koule, AABB, OBB, k-DOPs, QuSPO, ..
- **časoprostorové meze, 4D prostor**
 - přesné výpočty prvního kontaktu
- **Voronoi diagramy**
 - udržování seznamu „nejbližších primitiv“, využití časové koherence

BV hierarchie obecně

Složnost algoritmu na detekci kolize:

$$T = N_V \times C_V + N_P \times C_P + N_U \times C_U + C_0$$

- T .. celková složitost testu v celé scéně (pro jeden čas)
- N_V .. počet testovaných BV párů
- C_V .. složitost testu jednoho BV páru
- N_P .. počet testovaných párů geometrických primitiv
- C_P .. složitost testu jednoho páru geometrických primitiv
- N_U .. počet aktualizovaných uzlů hierarchie
- C_U .. složitost aktualizace jednoho uzlu hierarchie
- C_0 .. složitost jednorázového předzpracování stromu

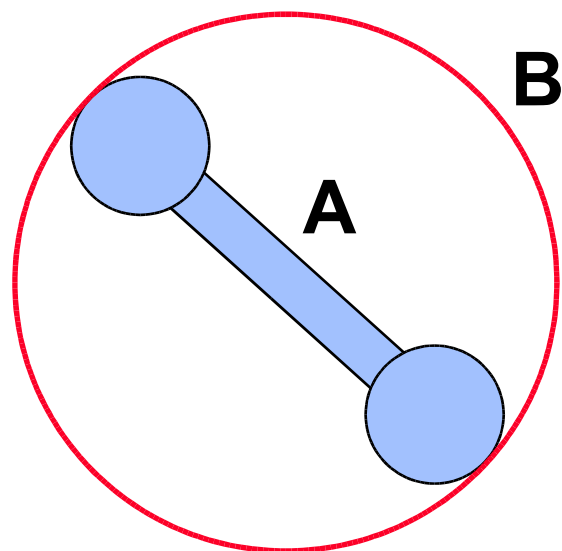
Další vlastnosti BV hierarchie

- čas potřebný k **předzpracování datové struktury**
 - výstavba stromu na úplném začátku
 - složitost jeho přestavby v případě velmi dynamických scén (interaktivní simulace, hry)
- **spotřeba paměti**
 - důležitá při velkých 3D modelech
 - některé hry mají dnes v databázi řádově 10^6 objektů, každý se může skládat ze stovek trojúhelníků...

Výběr konkrétního typu BV

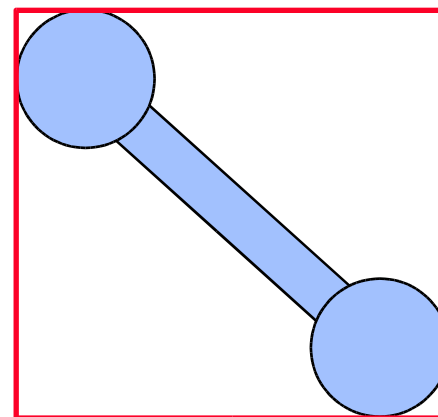
- **co nejtěsnější aproximace** originálního modelu
 - zmenšení N_V , N_P , N_U
- **rychlý test mezi obalovými tělesy**
 - zmenšení C_V
- **rychlá aktualizace** při animaci
 - co nejmenší C_U
- **efektivní konstrukce** s minimální paměťovou spotřebou

Různě efektivní obaly



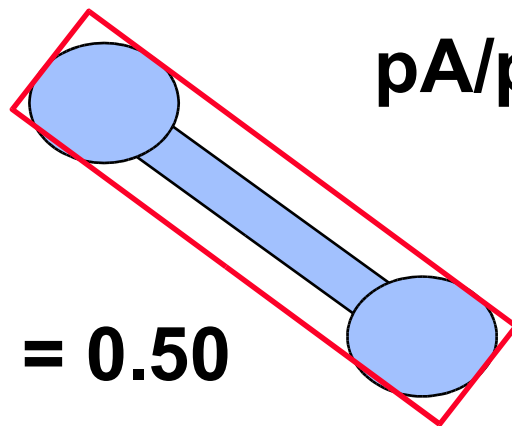
$$pA/pB = 0.20$$

„Sphere“



$$pA/pB = 0.23$$

„AABB“



$$pA/pB = 0.50$$

„OBB“

Příklady

- **konvexní obaly**
 - minimální \mathbf{N}_V , \mathbf{N}_P , \mathbf{N}_U ale velmi velké \mathbf{C}_V , \mathbf{C}_U
- **obalové koule**
 - nulové \mathbf{C}_U , velmi malé \mathbf{C}_V , ale velké \mathbf{N}_V
- **AABB** (osově orientované kvádry)
 - velmi malé \mathbf{C}_V , ale velké \mathbf{C}_U a též \mathbf{N}_V
- **OBB** (libovolně orientované obalové kvádry)
 - malé \mathbf{N}_V , \mathbf{N}_P , trochu větší \mathbf{C}_V , \mathbf{C}_U

Orientované obalové kvádry OBB

- v Ray-tracingu se používají od 80. let
 - jsou známy i některé teoretické výsledky: konstrukce optimálního OBB v čase $O(n^3)$ /1985/
- **2D** varianta OBB hierarchie je známa pod jménem „**strip tree**“
 - rychlý výpočet průsečíků křivek v rovině
 - Ballard 1981

Konstrukce sub-optimálních OBB

- výpočet **konvexního obalu** daného tělesa
 - funguje i pro „triangle soups“
- **těžiště** konvexního obalu M
 - počítám jen s vrcholy nebo integruji přes celé stěny
- **kovarianční matice** tří složek souřadnic
 - vrcholy nebo těžiště obalových stěn
 - též mohu integrovat přes celé obalové stěny (ne příliš složitý vzorec)

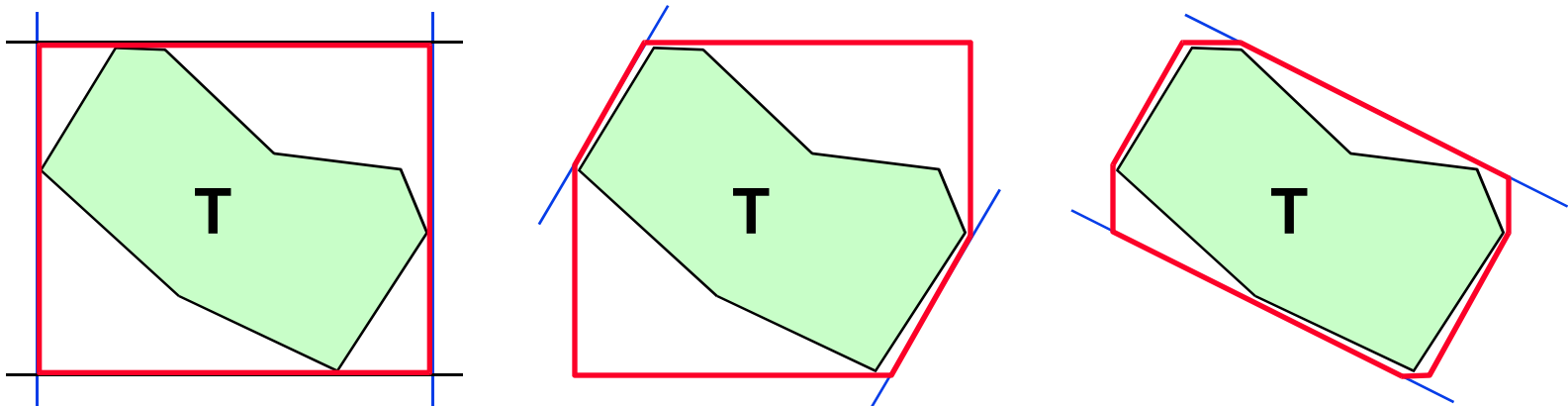
Konstrukce sub-optimálních OBB

- **výpočet vlastních vektorů kovarianční matice**
 - matice má rozměr **3 x 3**
- **dva vlastní vektory** definují směr **minimálního a maximálního rozptylu**
 - výhodné pro těsné obalení originálního tělesa
- **při výpočtu hierarchie** se mohou částečné součty uchovávat v cache paměti

K-DOPs

- výhodné obalové těleso pro **konvexní objekty**
- průnik několika **pásů** (daná množina **k** směrů)
 - pás je omezen dvěma rovnoběžnými rovinami
 - nutnost efektivního výpočtu konstant **d** a **D**:

$$\mathbf{d} = \min_{[x,y,z] \in T} \{ \mathbf{ax} + \mathbf{by} + \mathbf{cz} \}, \quad \mathbf{D} = \max_{[x,y,z] \in T} \{ \mathbf{ax} + \mathbf{by} + \mathbf{cz} \}$$



Literatura I

- [Cohen95] J. Cohen, M. Lin, D. Manocha, M. Ponamgi: *I-collide: An Interactive and Exact Collision Detection System for Large-Scale Environments*, I-collide
- [Gottschalk96] S. Gottschalk, M. C. Lin, D. Manocha: *OBB-Tree: A Hierarchical Structure for Rapid Interference Detection*
- [He99] Taosong He: *Fast Collision Detection Using QuOSPO Trees*
- [Held95] M. Held, J. Klosowski, J. Mitchell: *Evaluation of collision detection methods for virtual reality fly-throughs*

Literatura II

- [Hubbard95] Philip M. Hubbard: *Collision Detection for Interactive Graphics Applications*, PhD thesis, brown.edu
- [Klosowski98] J. Klosowski, M. Held, H. Sowizral, K. Zikan: *Efficient Collision Detection Using Bounding Volume Hierarchy of k -DOPs*
- [Klosowski98PhD] James T. Klosowski: *Efficient Collision Detection for Interactive 3D Graphics and Virtual Environments*, PhD thesis
- [Konecny98] P. Konecny: *Bounding volumes in computer graphics*, Diploma thesis, MU Brno

Literatura III

- [Krishnan97b] S. Krishnan, A. Pattekar, M. Lin, D. Manocha: *Spherical Shell: A Higher Order Bounding Volume for Fast Proximity Queries*
- [Krishnan98] S. Krishnan, M. Gopi, M. Lin, D. Manocha, A. Pattekar: *Rapid and Accurate Contact Determination between Spline Models using ShellTrees*
- [Lin96] M. C. Lin, J. Cohen, S. Gottschalk: *Collision Detection: Algorithms and Applications*, RAPID
- [Ponamgi95] M. Ponamgi, D. Manocha, M. Lin: *Incremental Algorithms for Collision Detection between Solid Models*