

# *Optimalizace BVH*

*Petr Kraus*

*[krauspe1@fel.cvut.cz](mailto:krauspe1@fel.cvut.cz)*

*Datové struktury pro počítačovou grafiku*

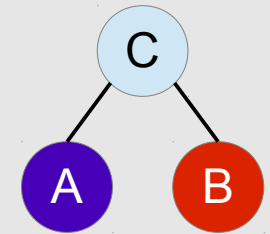
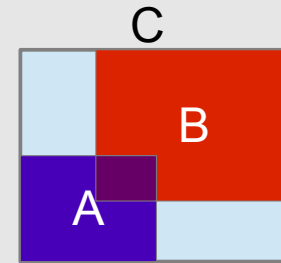
podle článku

## **Fast Insertion-Based Optimization of Bounding Volume Hierarchies**

[Jiří Bittner, Michal Hapala, Vlastimil Havran]

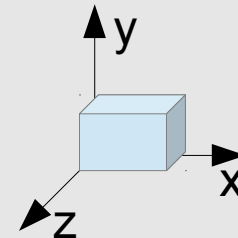
# Základní znalosti

- Bounding volume hierarchy
  - aka **BVH**, hierarchie obálek



- Axis-aligned bounding box

- aka **AABB**, osově zarovnaný obalový kvádr



- Surface area heuristic (SAH)

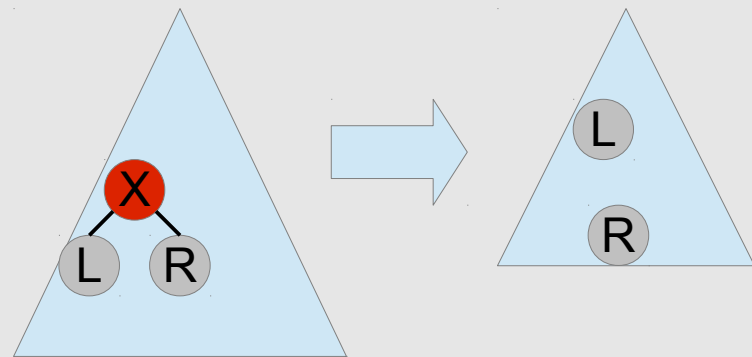
$$C(Tree) = \frac{C_{Tr} * \sum_{N \in Inner} SA(N) + C_I * \sum_{N \in Leafs} SA(N) * T_{count}}{SA(T)}$$

$$C(Tree) = const. * \sum_{N \in Inner} SA(N) + const.$$

# Algoritmus - úvod

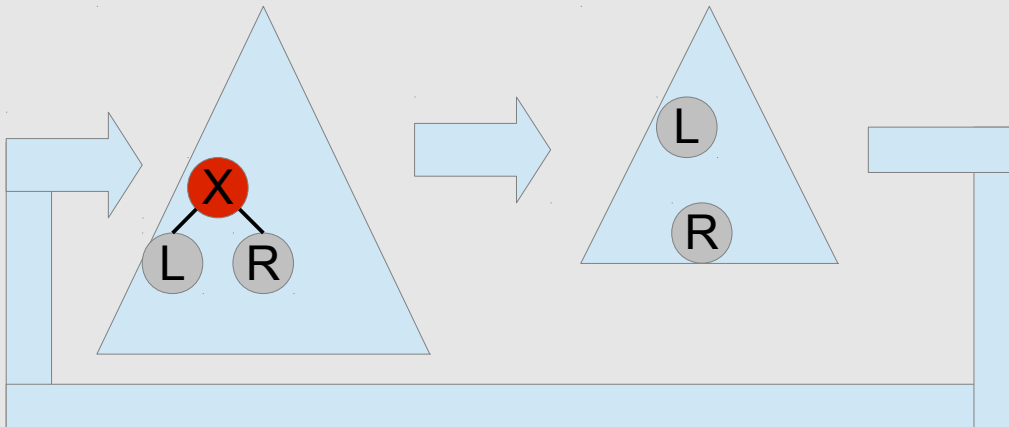
- Zlepšuje existující BVH
  - až 27% oproti SAH BVH
  - až 88% oproti prostorovému mediánu
- Rychlejší než simulované žíhání
- Lze řídit čas a kvalitu optimalizace

- Princip:
  - **vyjmutí** vybraných uzlů
  - jejich **lepší začlenění**
  - lze **opakovat** vícekrát



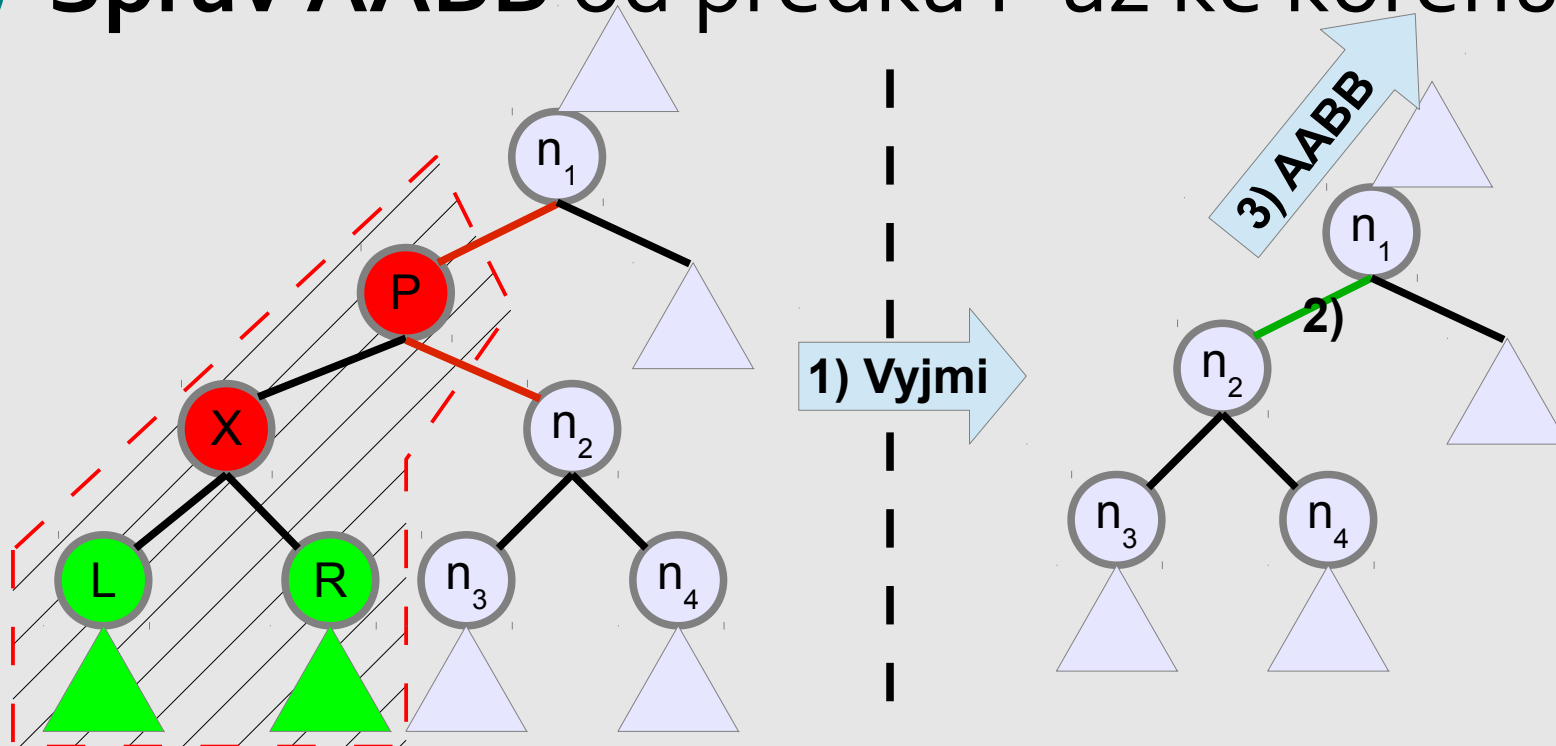
# Algoritmus – schéma

- (1) Opakuj až **ukončovací podmínka** splněna:
- (2) **Vyber** neefektivní vnitřní uzly
- (3) Každý vybraný uzel zpracuj:
  - (a) **Vyjmi** oba potomky uzlu
  - (b) **Nalezni** vhodné nové pozice
  - (c) **Začleň** potomky na nové pozice



# (3a) Vyjmutí potomků

- Pro vybraný uzel X:
  - 1) Vyjmi X, jeho předka(P) a potomky(L a R)
  - 2) Sprav konektivitu stromu
  - 3) Sprav AABB od předka P až ke kořenu



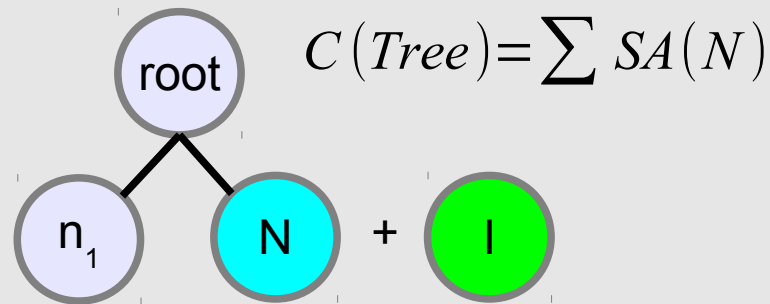
## *(3b) Nalezení nové pozice*

- Chceme vkládat **I**
  - Nejdříve ten s větší plochou z **L** a **R**
- Vybíráme ze **všech N** ve stromu pozici
  - Hledáme **uzel  $N_{best}$**  s **nejmenší cenou** pro **I**
- Vybíráme metodou **větví a mezí** od kořene
  - prořezávání větví

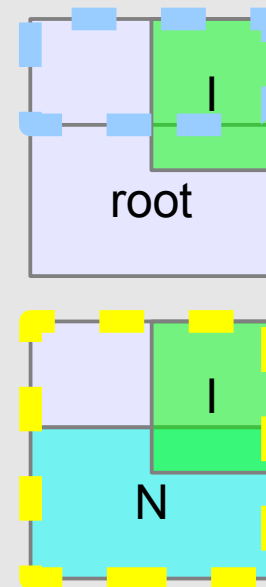
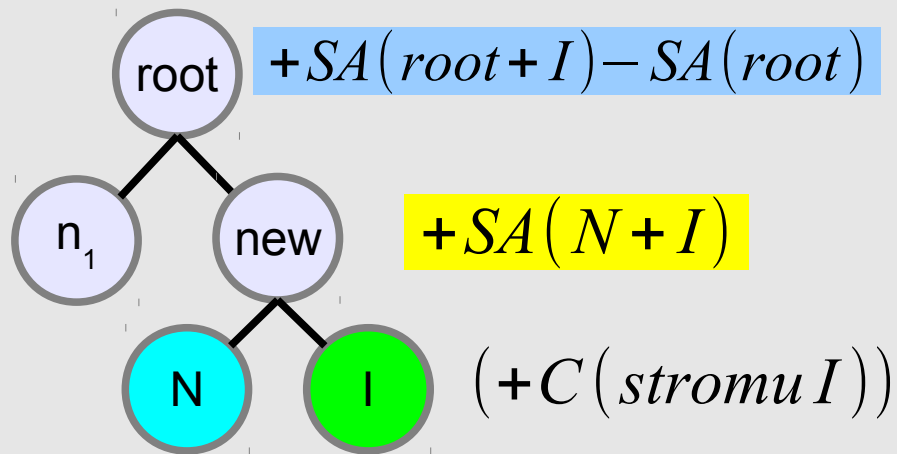


# (3b) Nalezení nové pozice - vizuálně

cena před přidáním:



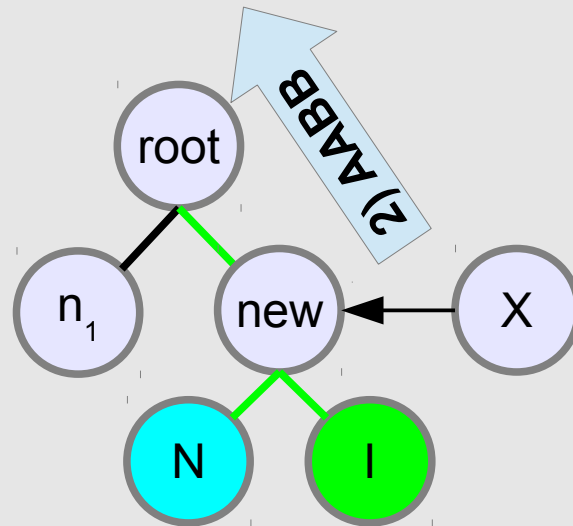
cena po přidání:





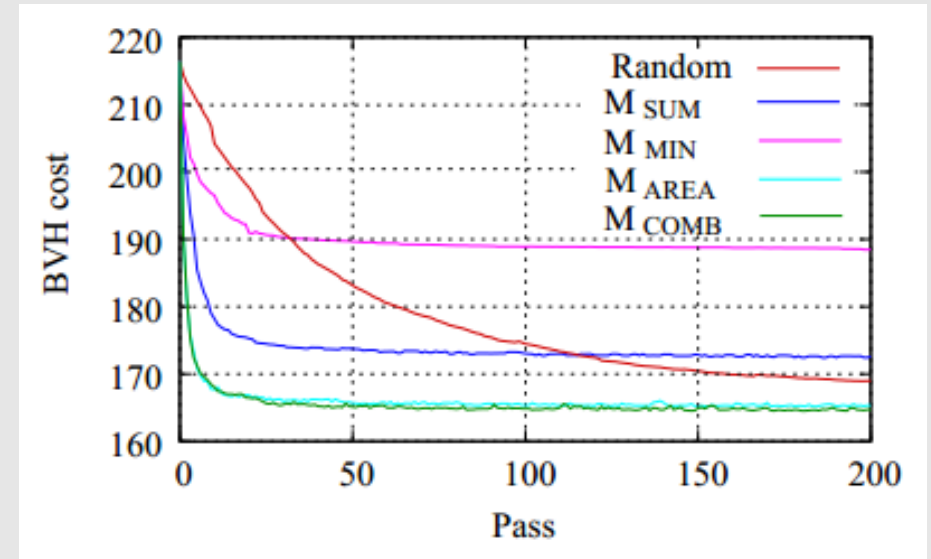
# (3c) Začlenění do stromu

- 1) Uzel  $N_{\text{best}}$  a  $I$  spojíme do nového **new**
  - recyklujeme k tomu  $X$  a  $P$
- 2) Spravíme AABB od **new** ke kořenu
  - jako při vyjímání



## (2) Výběr uzlů

- Kolik uzlů?
  - jeden
  - 1%
  - jinak?
- Podle čeho?
  - Náhodně
  - $M_{\text{AREA}} = \text{SA}(\text{Node})$
  - kombinace více měřítek



# *(1) Ukončovací podmínka*

- Konvergence ceny stromu
- Dosažení požadované ceny stromu
- Překročení časového limitu
- Překročení počtu iterací

# Implementace - návrh

## 1) Základní BVH

- top-down + prostorový medián

## 2) Optimalizace BVH přesouváním

- výběr podle SA(Node)
- výběr 1% uzlů stromu
- ukončení podle konvergence ceny

## 3) CPU raycaster

- primární + stínové paprsky
- stejné scény jako vy

# Implementace - zhodnocení

Scéna / počet ▲	C před	C po [-] a [%]	Čas stavby	Čas optimalizace [t] a [%]	Zlepšení času vykreslení[%]

- Grafy
  - počet iterací x cena
  - (počet iterací x čas iterace)

*FIN*

Připomínky a dotazy