

## CHO ĐỐI TƯỢNG 3D

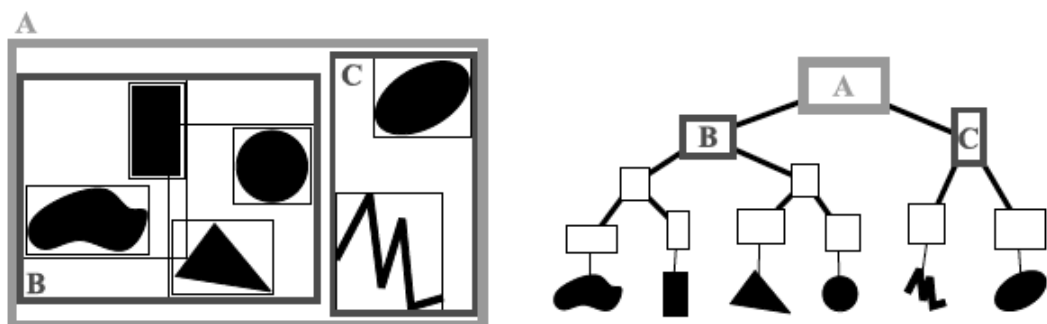
<sup>1</sup>Học viện Công nghệ Bưu chính Viễn thông, <sup>2</sup>Viện Công nghệ thông tin, <sup>3</sup>ĐH Thái Nguyên  
hoangnd@ptit.edu.vn

**Từ khóa** - hệ bao, tư đông, nhiều dạng hộp bao, nhân dạng và chạm

## I- GIỚI THIỆU

BVH hiện nay là một trong những phương pháp tiếp cận thể hiện tính thành công nhất trong các hệ thống hiện hành [[1]]. Thời gian tính toán cho các hệ thống này thể hiện độ ưu việt của BVH [[2]]:

$$\mathbf{T} = \mathbf{N}_v \times \mathbf{C}_v + \mathbf{N}_p \times \mathbf{C}_p$$



**Hình 1.** Ví dụ về một hệ bao sử dụng hình chữ nhật làm khối bao

**Cp:** Thời gian của phép thử cho một cặp các hình cơ bản.

Điều này chứng tỏ một hệ thống hoạt động sẽ dựa trên hai yếu tố: độ khít của hệ bao so với đối tượng ( $N_v$ ,  $N_p$ ) và độ đơn giản của phép thử chồng lấn trên một cặp hệ bao ( $C_v$ ).

Hệ bao khối cầu (Sphere) [[4]] và khối lập phương (AABB) [[3]] tạo ra phép thử chồng lấn đơn giản nhất. Trong khi đó, hệ bao khối chữ nhật (OBB) [[2]] và khối đa diện rời rạc có hướng (k-DOP) [[5]] cho biểu diễn khít nhất. Trong báo cáo này sẽ trình bày về việc ứng dụng hai loại khối biểu diễn để tối ưu cả về mặt độ khít của hệ bao và độ đơn giản của phép thử chồng lấn.

Beckmann [[3]] đưa ra giải thuật cho cây AABB, Palmer [[7]] và Hub-bard [[4]] đưa ra giải thuật cho cây khối cầu để giải quyết vấn đề đơn giản hóa. Trong khi đó Gottschalk [[2]] đưa ra giải thuật cho khối OBB còn Klosowski [[5]] đưa ra giải thuật cho khối đa diện k-DOP để giải quyết vấn đề về độ khít của hộp bao. Van den Bergen [[8]] đưa ra một phương thức đơn giản để phân tách các hộp chữ nhật OBB được biết đến với tên SAT lite. Giải thuật này chỉ sử dụng 6 trong số 15 hệ trục tọa độ so giải thuật gốc.

Trong báo cáo này, chúng tôi đề cập tới vấn đề xây dựng hệ bao (BVH) cho một đối tượng 3D dựa trên việc sử dụng nhiều dạng hộp bao với hai mục tiêu: giảm thời gian tính toán nhưng vẫn đạt được độ chính xác.

Phần còn lại của báo cáo được tổ chức như sau:

- Phần 2: Trình bày về Hệ bao (BVH)
- Phần 3: Trình bày về Kỹ thuật xây dựng hệ bao tự động với nhiều dạng hộp bao
- Phần 4: Thực nghiệm
- Phần 5: Kết luận

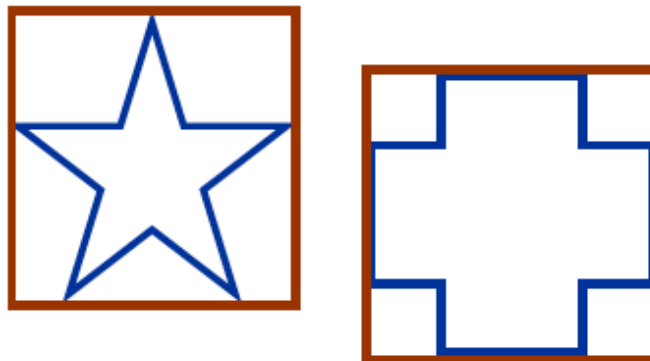
## II- HỆ BAO (BOUNDING VOLUME HIERARCHY)

### A. Hộp bao

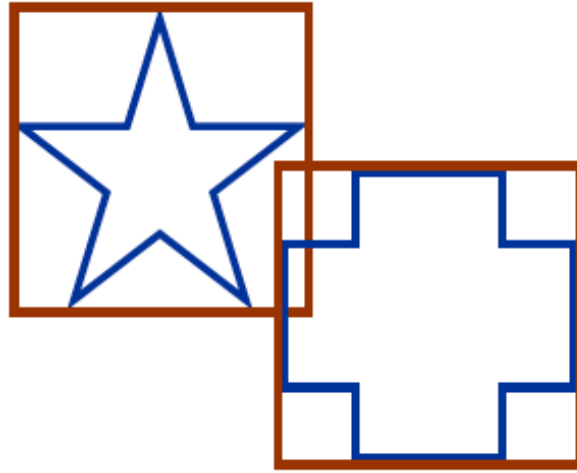
Đối với các đối tượng 3D, việc giải quyết các bài toán như nhận dạng va chạm, dò tia, ... cần phải xem xét đến bề mặt cũng như phần thể tích bên trong của đối tượng. Việc này trở nên phức tạp và rất tốn tài nguyên nếu đối tượng xem xét có hình dạng phức tạp. Để phân tích các tác động lên các đối tượng này, hộp bao được sử dụng. Thay vì việc cần phải xem xét toàn bộ đối tượng, hộp bao cho phép việc chỉ cần tính toán dựa trên các hình học đơn giản. Đối với các bài toán không yêu cầu độ chính xác quá cao, việc xem xét giới hạn ở phân tích bề mặt (3D) hoặc đường bao (2D) của hộp bao.

Tuy nhiên, cùng với độ đơn giản tính toán được giảm xuống, các bài toán có sử dụng hộp bao cần thừa nhận:

- Các phép tính chỉ dừng lại ở mức gần đúng
- Tính chính xác của các phép tính sẽ dựa trên độ khít của đường bao

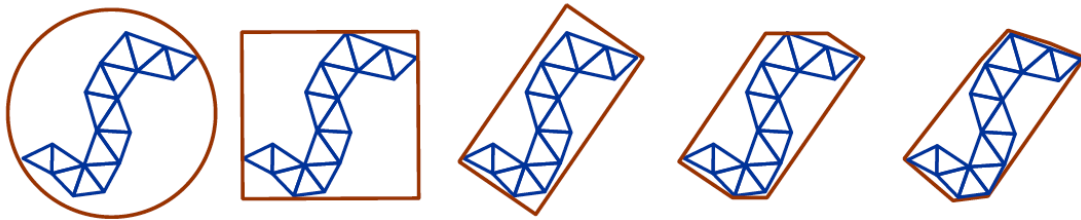


Hình 2. Không có chồng lấn hộp bao - Không có va chạm



**Hình 3.** Có chồng lẫn hộp bao - Có thể có va chạm

Hiện nay, để xây dựng hệ bao cho đối tượng, các dạng hộp bao thường được sử dụng gồm:

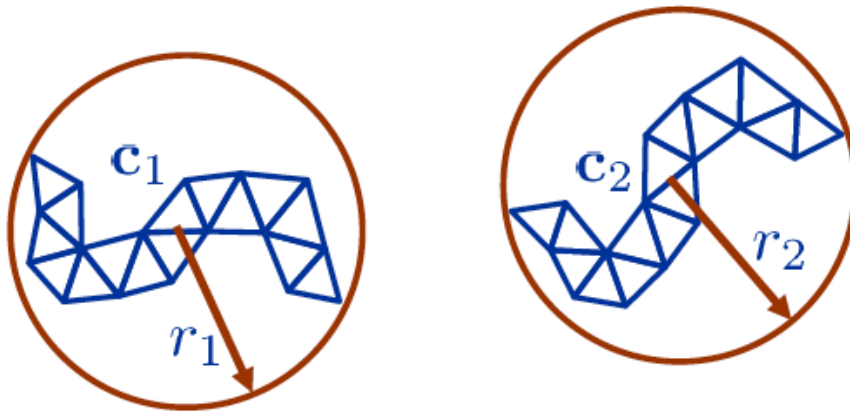


**Hình 4.** Các dạng hộp bao

- Hộp bao khối cầu: Sphere
- Hộp bao khối lập phương: AABB
- Hộp bao khối chữ nhật có hướng: OBB
- Hộp bao khối đa diện rời rạc có hướng: k-DOP
- Hộp bao khối lồi: convex hull

Hộp bao khối cầu: được biểu diễn bởi tâm (c) và bán kính khối cầu (r). Hai khối cầu không chồng lẫn lên nhau khi:

$$(C_1 - C_2) \cdot (C_1 - C_2) > (r_1 - r_2)^2$$

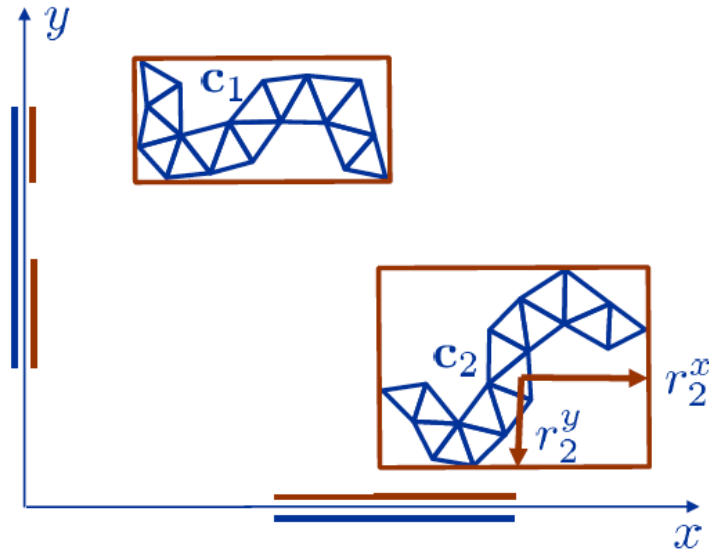


**Hình 5.** Va chạm giữa hai khối cầu

Hộp bao khối chữ nhật AABB: được biểu diễn bởi tâm hộp (c) và tham số chiều dài các cạnh (rx, ry, rz). Hai khối hộp lập phương không chồng lẫn lên nhau khi (xét trong miền không gian 2D):

$$|(c_1 - c_2) \begin{pmatrix} 1 \\ 0 \end{pmatrix}| > r_1^x + r_2^x$$

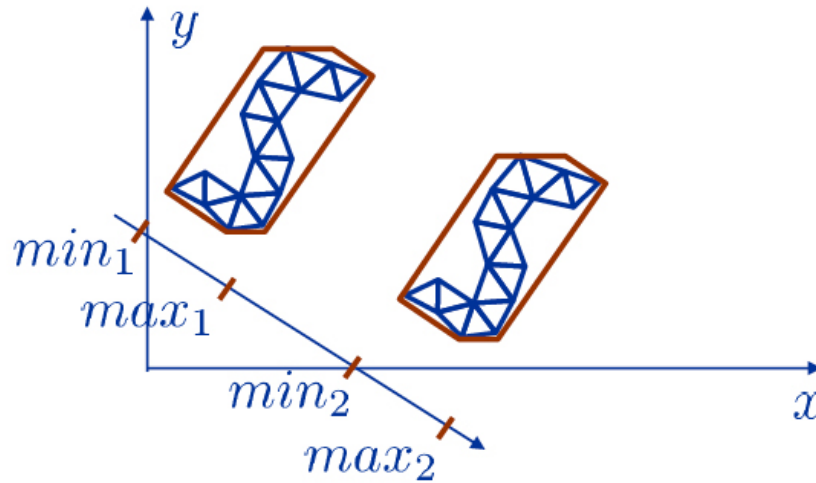
$$|(c_1 - c_2) \begin{pmatrix} 0 \\ 1 \end{pmatrix}| > r_1^y + r_2^y$$



Hình 6. Va chạm giữa hai khối hộp AABB

Hộp bao khối đa diện rời rạc có hướng: được xác định bởi hai tham số:  $k/2$  trung bình;  $k/2$  khoảng cách lớn nhất - nhỏ nhất. Như vậy nếu trong miền không gian 2D có thể coi AABB là 4-DOP, trong miền không gian 3D có thể coi AABB là 6-DOP. Hai cặp hộp đa diện sẽ không chồng lấn lên nhau khi (xét trong miền không gian 2D):

$$\exists \text{direction} : \max_1 < \min_2 \vee \min_1 < \max_2$$



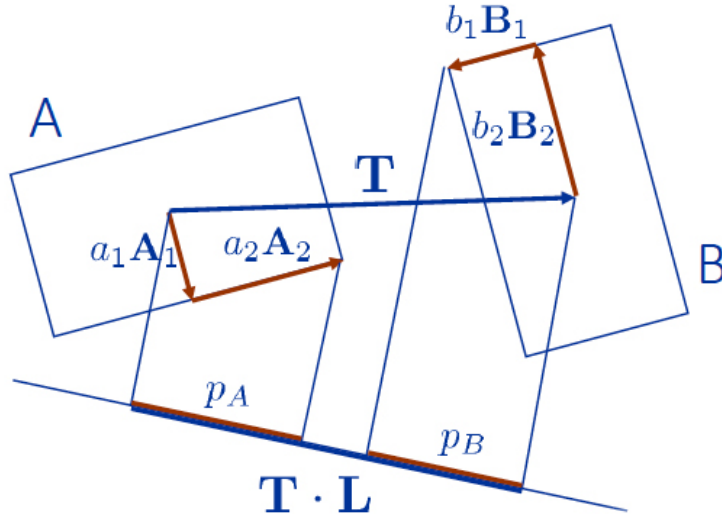
Hình 7. Biểu diễn khối OBB

Hộp bao khối chữ nhật có hướng OBB: Giống như khối hộp lập phương AABB nhưng có khả năng xoay. Bài toán xác định không chồng lấn đối với khối hộp OBB đã được nghiên cứu khá chi tiết:

- Trong miền không gian 2D: OBB được biểu diễn bởi các tham số:
  - $A1, A2, B1, B2$ : pháp tuyến vuông góc của hai đối tượng A và B
  - $a1, a2, b1, b2$ : số đo các cạnh của hai hộp
  - $L$ : pháp tuyến chỉ hướng
  - $T$ : Khoảng cách giữa A và B
  - $pA = a1A1L + a2A2L$
  - $pB = b1B1L + b2B2L$

- A và B không chồng lấn nhau khi:

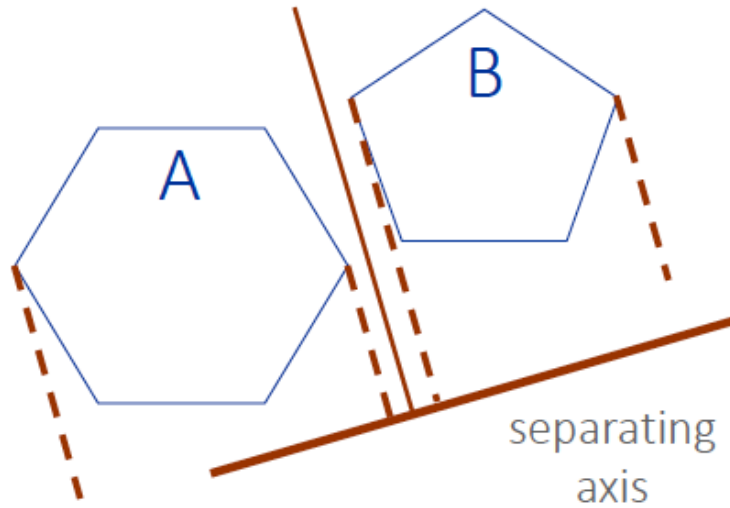
$$\exists L: T \cdot L > pA + pB$$



**Hình 8.** Xác định va chạm giữa hai khối OBB

Để xét hai đối tượng lỗi có chồng lấn lên nhau hay không, một trục tọa độ phân tách v sẽ được xác định giữa hai đối tượng. Đối với các đối tượng này một số các trục cần xem xét như sau:

- Trục song song với mặt trung bình của A
- Trục song song với mặt trung bình của B
- Trục song song với mặt cắt tại các góc của A và B

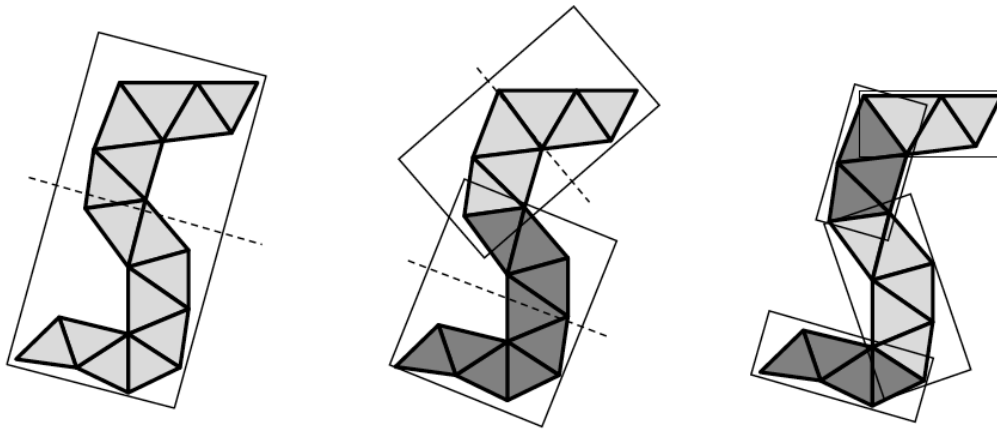


**Hình 9.** Xác định va chạm giữa hai khối đa diện

- Trong miền không gian 3D: Để xác định chồng lấn các trục cần xem xét gồm 15 trục để xác định được trục tọa độ phân tách.

### B. Hệ bao

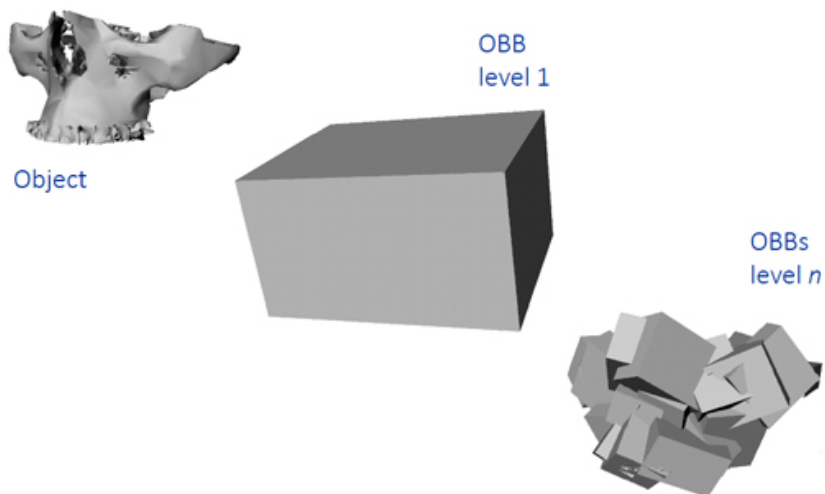
Là một cấu trúc dữ liệu dạng cây được xây dựng trên cơ sở phân tích các đối tượng được xem xét dựa trên cơ sở các hộp bao hình học. Tại các lá chứa các hình hình học cơ bản



**Hình 10.** Hệ bao xây dựng bởi các hộp bao

Đặc điểm của Hệ bao:

- Các nút trong một nhánh phải gần nhau hơn so với các nút khác. Càng xuống thấp thì các nút càng phải gần nhau hơn.
- Mỗi nút trong BVH cần có thể tích nhỏ nhất
- Tổng của các khối bao cần phải tối giản
- Các nút càng gần gốc thì càng quan trọng. Việc loại bỏ một nút gần gốc sẽ ảnh hưởng lớn hơn nhiều lần so với các nút ở xa.
- Thể tích trùng nhau của các nút đồng cấp phải tối giản.



**Hình 11.** Hệ bao xây dựng bởi hộp bao OBB

- Độ khít: Độ khít có thể tính toán qua thể tích. []

$$\tau = \frac{volume(B)}{\sum_{C \in C(B)} volume(C)}$$

- $C(B)$  là tập các nhánh con tại nút  $B$
  - $volume(B)$  là thể tích của hệ bao tại  $B$
  - $\tau$  là độ khít
- Giá trị của hệ bao
  - $H$  là hệ bao
  - $C(n)$  là tập các nhánh con tại nút  $n$
  - $cost$  là giá trị hệ bao

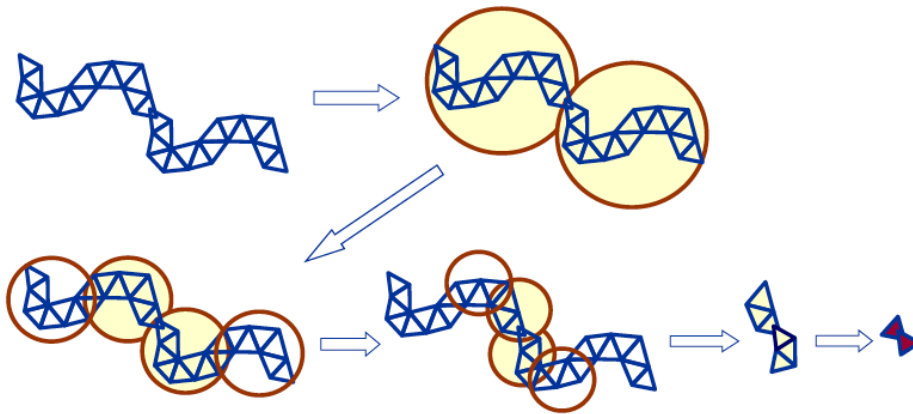
$$cost(H) = \sum_{n \in H} (C(n).volume(n))$$

Phương thức thiết lập cây:

- Từ trên xuống: Chia đầu vào thành hai (hoặc nhiều) nhánh, bao chúng lại, sau đó tiếp tục chia nhỏ các nhánh đến khi mỗi nhánh chỉ chứa một hình cơ bản. Phương pháp này cho phép tạo ra cây đơn giản nhưng không được ứng dụng nhiều trong thực tế.
- Từ dưới lên: Bắt đầu với các hình cơ bản tại các nhánh, sau đó cộng gộp dần để xây dựng thành đối tượng ban đầu. Phương pháp này khó thực hiện nhưng nhìn chung có thể tập hợp thành cây tốt hơn.
- Thêm vào: Hai phương pháp trên sử dụng tất cả các hình cơ bản trước khi tổ hợp thành cây. Phương pháp thêm vào cho phép không cần sử dụng tất cả các hình cơ bản. Cây ban đầu được xây dựng là một cây rỗng và được xây dựng dần bằng việc xác định cây nhỏ nhất.

Phương thức kiểm tra đối với cây:

- Nếu hộp bao trên một tầng nào đó của hệ bao bị chồng lấn, các nhánh con của nó cần được kiểm tra
- Tại các lá, việc kiểm tra thực hiện đối với các hình hình học cơ bản
- Loại bỏ các phần đối tượng không chịu tác động



Hình 12. Phân tích và chạm ảnh hưởng tới các phần tử của hệ bao

### III- KỸ THUẬT XÂY DỰNG HỆ BAO TỰ ĐỘNG VỚI NHIỀU DẠNG HỘP BAO

Giới hạn cho việc thực hiện kỹ thuật này như sau:

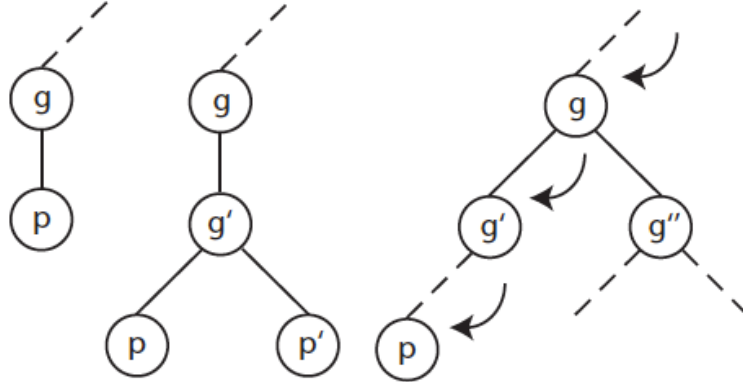
- Việc thực hiện được thực hiện trên hai vật thể rắn. Tính ưu việt của kỹ thuật được thể hiện qua việc cho hai vật thể rắn giống hệt nhau và chạm với nhau. Thời gian tính toán và chạm là tiêu chí để xem xét.
- Việc biểu diễn hệ bao đối tượng với nhiều dạng hộp bao sẽ giới hạn ở hai dạng hộp bao thuộc về mỗi phương hướng tối ưu.
- Một hệ bao với hai dạng hộp bao được lựa chọn, trong đó mỗi nút hộp bao thuộc hướng khác sẽ được tăng cường bởi một hộp bao hướng đơn giản.
- Phép thử với hộp bao hướng đơn giản sẽ được thực hiện trước để loại trừ các đối tượng ở xa.

#### A. Xây dựng hệ bao tự động

Việc xây dựng tự động hệ bao có thể được coi là tự động xây dựng cấu trúc dữ liệu hình cây mô tả hệ bao [[10]]. Phương thức chung để xây dựng một hệ bao có thể được miêu tả như sau: hệ bao được xây dựng trên cơ sở một cây dữ liệu các hộp bao. Trong đó các hộp bao là các hình đơn giản được sắp xếp khác nhau, bao phủ đối tượng cần xem xét. Các hộp này được đề cập đến ở **phần II**.

Một số giải thuật xây dựng hệ bao tự động đã được nghiên cứu:

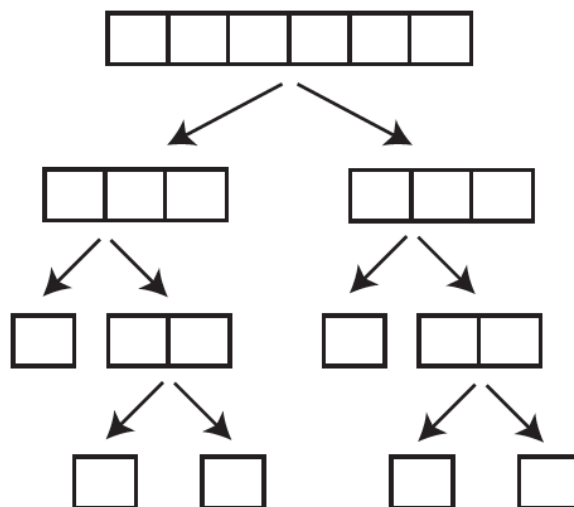
- Thêm dần: Giải thuật được đưa ra bởi Goldsmith [11]. Giải thuật được thiết lập dựa trên việc tính toán giá trị nhỏ nhất của cây khi thêm các hình cơ bản vào trong hệ. Khi một hình cơ bản  $p$  được thêm vào trong một hệ được phân chia, giải thuật sẽ sử dụng 3 luật:
  - $p$  có thể là nhánh con của một nhóm  $g$
  - $p$  có thể kết hợp với một hình cơ bản  $p'$  nhóm  $g'$ ,  $g'$  sẽ là một nhánh con của  $g$
  - $p$  có thể được thêm vào một nhóm  $g'$  thuộc nhóm đệ quy của  $g$



Hình 13. 3 Luật của thuật toán thêm vào.

Phương pháp này có thể được sử dụng để tạo một hệ bao xấp xỉ tuy nhiên nó có một số hạn chế. Hệ được tạo ra dựa trên yêu cầu thêm vào của các nút. Và yêu cầu này là không mong muốn do phải dựa trên cảm quan của người xây dựng hệ bao. Trong một số trường hợp giá trị của cây sẽ không tối ưu và mỗi nhóm mới chỉ chứa hai hình cơ bản. Điều này được cải thiện hơn trong thuật toán được đưa ra bởi Haber [12]: sử dụng hai cách tiếp cận:

- Thêm lại thành công: Loại bỏ những nút không tốt và thêm lại chúng vào hệ bao
- Giới hạn các nhóm xấu: Tìm các nhóm không tốt và cố gắng chia chúng ra.
- Chia nhỏ: Thuật toán này được xây dựng bởi Muller [13]. Thuật toán chia nhỏ một tập hợp các hình cơ bản một cách đệ quy thành hai tập con không trùng phần tử. Việc này được dừng lại khi đạt đến ngưỡng. Thuật toán sẽ thực hiện như sau: Cây hệ bao được xây dựng bởi việc sắp xếp các hình cơ bản theo các trục tọa độ chính và lấy mốc là tâm của các hình cơ bản. Sau đó chức năng lựa chọn giá trị nhỏ nhất của cây hoạt động trên việc xem xét tất cả các điểm phân chia có thể. Thuật toán sẽ tiếp tục chia đến khi các cây chứa toàn các hình cơ bản tại các lá.

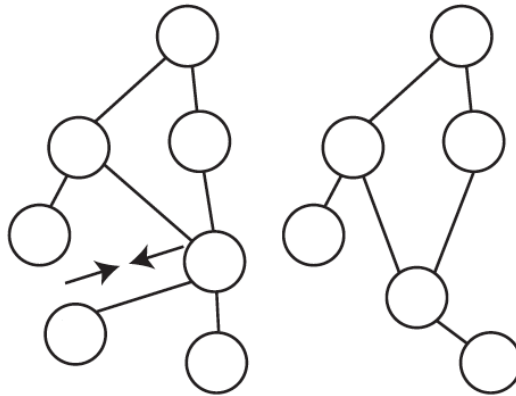


Hình 14. Xây dựng cây bằng cách phân chia dọc theo một trục tại các điểm có giá trị nhỏ nhất.

Giải thuật này cũng được Gottschalk [14] sử dụng cho hệ bao sử dụng hộp bao OBB. Trong đó, việc chia nhỏ sẽ tiến hành dọc theo trục dài nhất, sử dụng các điểm trung tâm.

Điểm hạn chế duy nhất của giải thuật này là chỉ xây dựng được các hệ bao nhị phân. Tuy nhiên có thể khắc phục bằng cách chia nhiều lần tại cùng mỗi cấp. Độ cân bằng của cây phụ thuộc và chức năng giá trị được sử dụng.

- **Kết hợp:** Giải thuật được xây dựng bởi Erleben [15] và có thể thấy được áp dụng trong OpenTissue [16]. Giải thuật này bắt đầu với việc xây dựng cấu trúc đồ thị dữ liệu, trong đó mỗi nút thuộc đồ thị liên quan đến các hình cơ bản và các đỉnh có quan hệ lân cận. Một đỉnh trong đồ thị nghĩa là hai nút trong hệ bao có thể kết hợp tốt với nhau. Các đỉnh được xác định bằng một chức năng phỏng đoán trong đó phỏng đại hộp bao cơ bản và ghi nhận va chạm. Một va chạm có nghĩa là một đỉnh giữa hai đồ thị nút vừa va chạm cần được thêm vào đồ thị.



**Hình 15.** Một đỉnh sụp đổ thành một nút

Việc này được lặp đi lặp lại cho đến khi một nút duy nhất tồn tại. Sau khi một đỉnh sụp đổ trong đồ thị, các nút thuộc hệ bao được kết hợp thành một nhóm mới khi một trong hai điều kiện sau thỏa mãn:

- Đồ thị nút bao phủ lượng lớn hơn một nhánh cố định.
- Có ít đỉnh hơn trong một đồ thị so với một nhánh cố định.

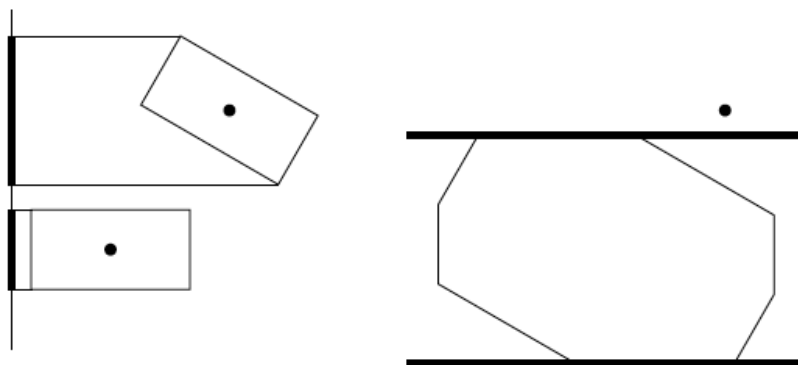
### **B. Lựa chọn hộp bao phù hợp**

Như đã trình bày ở trên, việc xây dựng hệ bao đối tượng có thể thông qua các phương pháp chính là: sử dụng hệ bao cầu (Sphere); hệ bao hộp chữ nhật (AABB); hệ bao hộp chữ nhật có hướng (OBB); và hệ bao đa diện có hướng rời rạc (k-DOP).

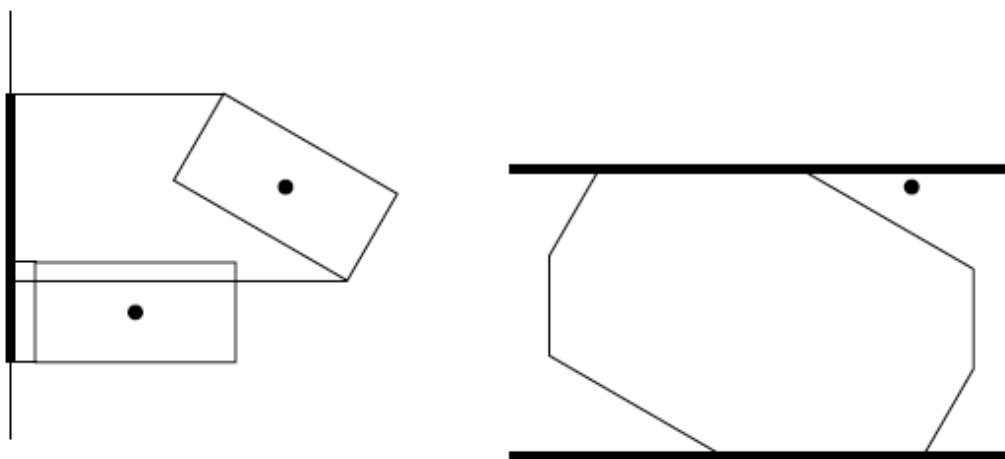
Để tận dụng lợi thế của hai dạng hộp bao: AABB, Sphere - đơn giản; OBB, k-DOP - chính xác, có thể xây dựng một cây hệ bao được xây dựng bằng nhiều dạng hộp bao trên mỗi nút. Trong đó, tại mỗi nút sẽ có 1 hộp bao dạng đơn giản và 1 hộp bao dạng chính xác.

Trong tài liệu này sẽ lựa chọn sử dụng hai dạng hộp bao: AABB và OBB để xây dựng cây hệ bao cho đối tượng. Cấu trúc cây cơ bản sẽ được xây dựng dựa trên cấu trúc cây OBB được đưa ra bởi Gottschalk. Với mỗi nút trên cây OBB được xây dựng, cấu trúc hai hộp bao sẽ được xây dựng bao gồm thêm một hộp bao AABB bao các thành tố của mặt phẳng tại nút đó.

Có hai phương thức để xây dựng hộp bao AABB trong trường hợp này. Phương thức thứ nhất sẽ tìm ra hộp bao AABB nhỏ nhất cho đối tượng. Phương thức thứ hai sẽ đặt tâm của hộp AABB trùng với tâm của hộp OBB. Phương thức thứ hai sẽ cho giải thuật đơn giản hơn và việc tính toán sẽ nhanh hơn. Trong khi đó phương án thứ nhất sẽ cho hộp bao AABB khít hơn đối với đối tượng. Theo một số thực nghiệm việc chọn khối hộp AABB khít sẽ cho kết quả của các phép thử tốt hơn.



Hình 16. Hai hộp bao không chồng lấn



Hình 17. Hai hộp bao chồng lấn

Bài kiểm tra cho việc phân tách nút đối với cây hệ bao hai dạng hộp bao sẽ được thực hiện như sau: Hệ hộp bao AABB sẽ được kiểm tra trước, nếu chúng cần phải chia nhỏ thì hệ bao chung sẽ chia nhỏ. Nếu hệ hộp bao AABB bị chồng lấn, khi đó hệ hộp bao OBB sẽ được xem xét tiếp theo.

Những ưu điểm của phương pháp xây dựng hộp bao này gồm:

- Tăng cường độ khít của hộp bao so với các phương pháp AABB đơn lẻ. Điều này đạt được do độ khít của hộp OBB tốt hơn so với hộp AABB
- Độ phức tạp của phép thử được giảm bớt so với phương pháp sử dụng hộp OBB. Do chỉ phải thực hiện phép thử với hệ hộp AABB trước, nếu xảy ra chồng lấn thì mới cần xét tiếp đến hệ hộp OBB nên số lượng tính toán của phương pháp kép sẽ giảm thiểu.

### C. Ưu điểm và hạn chế

Ưu điểm chính của kỹ thuật là việc không làm giảm độ chính xác của các phép thử do sử dụng hệ bao đảm bảo chính xác (OBB) làm cơ sở và khả năng tăng tốc tính toán do sử dụng hệ bao đảm bảo tính đơn giản (AABB) để tính toán trước, khi va chạm xảy ra tại nhánh nào thì mới khoanh vùng để tính chính xác.

Hạn chế của kỹ thuật là thời gian xây dựng hệ bao sẽ tăng lên nhiều so với phương pháp sử dụng hệ bao một dạng hộp bao. Ngoài ra do có hai dạng hộp bao trên một vật thể nên kích thước của đối tượng được xem xét cũng sẽ tăng lên.

### D. Xây dựng thuật toán

Các bước xây dựng thuật toán có thể được mô tả như sau:

- Bước 1: Xây dựng cây dữ liệu hệ bao theo phương pháp xây dựng hệ bao tự động sử dụng cho dạng hộp bao là AABB theo giải thuật của Gottschalk
- Bước 2: Tại mỗi nút trên cây đã xây dựng tái tạo một cây mới, có cấu trúc cây giống cây cũ. Dạng hộp bao được sử dụng sẽ được thay thế bằng OBB.
- Bước 3: Giải thuật được xây dựng sẽ tính toán dựa trên cơ sở việc phát hiện va chạm xảy ra với hệ bao
  - Nếu không xảy ra va chạm. Hệ bao cho đối tượng sẽ là hệ bao sử dụng dạng hộp bao là AABB

- Nếu xảy ra va chạm tại một nút nào đó thuộc hệ bao. Hệ bao cho đối tượng sẽ là hệ bao sử dụng dạng hộp bao là OBB

## IV- THỰC NGHIỆM

Việc thực nghiệm thể hiện kết quả cho việc tính toán thời gian xử lý được áp dụng cho các dạng bề mặt khác nhau, với các cấu hình khác nhau.

**Bảng 1.** Bảng so sánh thời gian xử lý (s)

|         |    | RAPID   | Dual    |
|---------|----|---------|---------|
| Mẫu thử | 0% | 27.2540 | 20.6053 |
|         | 1% | 14.0696 | 10.1924 |
|         | 2% | 8.6457  | 5.8939  |
|         | 3% | 6.2860  | 4.0741  |
|         | 4% | 4.9193  | 3.0381  |
|         | 5% | 4.0032  | 2.3816  |

Mẫu thử: Sử dụng mô hình Phật đi lạc làm mẫu thử. Mẫu thử bao có hệ lưới bao gồm 15.536 tam giác. Va chạm xảy ra với hai đối tượng giống nhau sẽ có 229,824 cách cấu hình vị trí và hướng mẫu thử. Cách cấu hình vị trí và hướng mẫu thử được đưa ra bởi Trenkel [17]. Trong đó sử dụng 6 dạng khoảng cách khác nhau: 0%, 1%, 2%, 3%, 4% và 5% cho kích thước mẫu thử đưa vào. Mỗi khoảng cách được xác định bởi bán kính của hộp bao.

Giải thuật RAPID: Giải thuật cho phép nhận dạng va chạm trên cơ sở sử dụng hộp bao OBB, có thể được download trên trang web: <http://www.cs.unc.edu/~geom/OBB/OBBT.html>. Trên cơ sở thay đổi mã nguồn mở của giải thuật này chúng tôi đã xây dựng giải thuật cho việc nhận dạng va chạm sử dụng hai dạng hộp bao.

Dựa trên kết quả có thể dễ dàng nhận ra, việc sử dụng giải thuật với hai dạng hộp bao sẽ tiết kiệm thời gian hơn so với việc sử dụng giải thuật RAPID.

## KẾT LUẬN

Xây dựng hệ bao BVH tự động cho một đối tượng 3D trong các bài toán tính toán va chạm là cách tiếp cận thể hiện nhiều ưu điểm. Việc xây dựng BVH cho đối tượng thường theo mô hình trên xuống, dưới lên hoặc thêm vào; với một dạng hộp bao cụ thể. Báo cáo này đề xuất một kỹ thuật xây dựng BVH dựa trên việc sử dụng nhiều dạng hộp bao khác nhau phù hợp với thực tế hoạt động của đối tượng. Kỹ thuật đã được thử nghiệm với hai dạng hộp bao và tỏ ra hiệu quả đối với các mô hình đối tượng 3D được xây dựng theo phương pháp liên tục.

## TÀI LIỆU THAM KHẢO

- [1]. Akenine-Moller, T., Hains, E.: Real-Time Rendering. A K Peters, 2002
- [2]. Gottschalk, S., Lin, M.C., Manocha, D.: OBB-Tree: a hierarchical structure for rapid interference detection. In: ACM SIGGRAPH 1996, pp. 171–180, 1996.
- [3]. Beckmann, N., Kriegel, H.-P., Schneider, R., Seeger, B.: The R-Tree: an efficient and robust access method for points and rectangles. In: ACM SIGMOD Conf. on the Management of Data, pp. 322–331, 1990.
- [4]. Hubbard, P.M.: Collision detection for interactive graphics applications. IEEE Trans. on Visualization and Computer Graphics 1(3), 218–230, 1995.
- [5]. Klosowski, J.T., Held, M., Mitchell, J.S.B., Sowizral, H., Zikan, K.: Efficient collision detection using bounding volume hierarchies of k-DOPs. IEEE Trans. on Visualization and Computer Graphics 4(1), 21–37, 1998.
- [6]. Lin, M.C., Gottschalk, S.: Collision detection between geometric models: a survey. In: Proc. IMA Conference on the Mathematics of Surfaces, pp. 37–56, 1998.
- [7]. Palmer, I., Grimsdale, R.: Collision detection for animation using sphere-trees. Computer Graphics Forum 14(2), 105–116, 1995.
- [8]. Van den Bergen, G.: Efficient collision detection of complex deformable models using AABB trees. J. Graphics Tools 2(4), 1–14, 1997.

- [9]. Herman J. Haverkort, Introduction to bounding volume hierarchies. PhD Thesis Chapter 1, 2004.
- [10]. Jepprey Goldsmith, John Salmon, Automatic creation of Object Hierarchy for Ray tracing. IEEE CG&A, 1987.
- [11]. K. Erleben, J. Sporring, K. Henriksen, and H. Dohlmann. Physics-Based Animation . Charles River Media, 2005.
- [12]. J. Goldsmith and J. Salmon. Automatic Creation of Object Hierarchies for Ray Tracing . IEEE CGA, 1987.
- [13]. J. Haber, M. Staminger, and H. Seidel. Enhanced Automatic Creation of Multi- Purpose Object Hierarchies . IEEE CGA, 2000.
- [14]. G. Müller, S. Schafer, and D. W. Fellner. Automatic Creation of Object Hierarchies for Radiosity Clustering . Technical Report TUBS-CG-1999-06, TU Braunschweig, 1999.
- [15]. K. Erleben. An Introduction to Approximating Heterogeneous Bounding Volume Hierarchies . Technical Report DIKU-TR-02/04, DIKU, 2002.
- [16]. Opentissue: Opensource Project, Physics-Based Animation and Surgery Simulation. <http://www.opentissue.org>.
- [17]. Trenkel, S., Weller, R., Zachmann, G.: A Benchmarking Suite for Static Collision Detection Algorithms. In: International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (WSCG), 2007.