**So sánh thuật toán Simulated Annealing và AND-OR Search**

| **Tiêu chí** | **Simulated Annealing** | **AND-OR Search** |
| --- | --- | --- |
| **Đầu vào (Input)** | - Bài toán tối ưu hóa (như bài toán 15-Queens, TSP, Sudoku, v.v.)  - Hàm đánh giá chi phí hoặc hàm fitness  - Thông số nhiệt độ, tỉ lệ làm mát (cooling schedule) | - Bài toán tìm kiếm trong không gian trạng thái có các hành động không xác định (MDP, trò chơi, lập kế hoạch AI, v.v.)  - Đồ thị không gian trạng thái |
| **Đầu ra (Output)** | - Một nghiệm tối ưu gần đúng (không đảm bảo tối ưu toàn cục) | - Một chính sách (policy) hoặc một tập hợp các hành động điều kiện |
| **Tốc độ (Speed)** | - Thường nhanh với các bài toán tối ưu hóa lớn  - Hiệu suất phụ thuộc vào cách giảm nhiệt độ | - Có thể chậm hơn do phải xử lý nhiều nhánh rẽ khi gặp các hành động không xác định |
| **Tốn bộ nhớ (Memory Usage)** | - Ít bộ nhớ hơn do không cần lưu toàn bộ cây tìm kiếm  - Chỉ lưu trạng thái hiện tại và một số giá trị trung gian | - Tốn bộ nhớ hơn do phải duy trì cây AND-OR lớn và lưu trữ nhiều trạng thái để xử lý không xác định |
| **Tính đầy đủ (Completeness)** | - Không đầy đủ: có thể không tìm thấy nghiệm nếu nhiệt độ giảm quá nhanh hoặc rơi vào cực trị địa phương | - Đầy đủ: nếu trạng thái đích có thể đạt được, thuật toán sẽ tìm được giải pháp |
| **Tính tối ưu (Optimality)** | - Không đảm bảo tối ưu toàn cục  - Có thể tìm được nghiệm tốt nhưng không chắc chắn là tốt nhất | - Có thể tìm được giải pháp tối ưu nếu triển khai đúng thuật toán |
| **Ứng dụng (Applications)** | - Bài toán tối ưu hóa không gian trạng thái lớn (ví dụ: TSP, xếp lịch, điều chỉnh mạng neuron, 15-Queens, Sudoku, v.v.) | - Lập kế hoạch trong môi trường không xác định (ví dụ: AI game, hệ thống quyết định, robot tự hành, v.v.) |

Nhận xét chung:

* Simulated Annealing phù hợp cho bài toán tối ưu hóa với không gian trạng thái lớn và phức tạp, cần tìm nghiệm tốt trong thời gian ngắn mà không yêu cầu tối ưu tuyệt đối.
* AND-OR Search phù hợp cho bài toán tìm kiếm trong môi trường không xác định, giúp tìm ra giải pháp có điều kiện thay vì một đáp án cố định.

Nếu em đang làm về bài toán 15-Queens, thì Simulated Annealing là lựa chọn phù hợp hơn. Còn nếu làm bài toán về tìm kiếm trong môi trường không xác định (AI planning, game AI, v.v.), thì AND-OR Search mới thực sự hữu ích.