

Báo cáo bổ sung Đồ án tốt nghiệp sau phản biện.

Sinh viên: Nguyễn Đình Tuấn Anh

Lớp: KSTN-CNTT K61.

Phần 1: Cách xác thực công thức tính kỳ vọng diện tích trong thuật toán:

Em cũng rất lo ngại rằng các công thức tính toán của mình có sai sót nên em đã tìm cách tự kiểm tra xem thuật toán của mình có chạy đúng hay không. Để xác minh điều này, em đã tiến hành tính bằng 2 cách như sau:

Cách 1 là em sử dụng phương pháp tích phân gần đúng như đã trình bày trong đồ án.

Cách 2 là em sử dụng thủ tục Monte-Carlo, cụ thể như sau:

- Tạo ngẫu nhiên 1000 mẫu phân phối Poisson trên bề mặt một tuyến đường.
- Mỗi mẫu này bao gồm k điểm có tọa độ (x, y) trên bề mặt đường (k có thể khác nhau trong mỗi mẫu).
- Với mỗi mẫu này, em gieo ngẫu nhiên 1000 điểm trên toàn bộ vùng cần theo dõi.
- Tiến hành đếm tổng số điểm nằm trong các đường tròn có tâm là một trong k điểm của mẫu và bán kính là R_0 (bán kính bao phủ của cảm biến chất lượng không khí).

Từ đó xác định được giá trị kỳ vọng gần đúng của diện tích vùng được theo dõi.

Em thử nghiệm trên 1 làn đường với chiều dài 10km và chiều rộng 25m.

Kết quả cụ thể em thu được tính bằng phần trăm diện tích bao phủ như sau:

Phương pháp $\lambda \left(\frac{\text{điểm}}{m^2} \right)$	Phương pháp tích phân	Thủ tục Monte-carlo
0.003	94.5027%	94.593%
0.001	87.5076%	86.9168%
0.0005	75.009%	74.4598%
0.0001	28.5632	28.5754%

Kết quả thu được luôn gần tương tự với nhau điều đó chứng minh rằng công thức tính toán em xây dựng và lập trình hoàn toàn đáng tin cậy.

Phần 2: Mô phỏng kết quả thí nghiệm

Ở phần này em xin trình bày kết quả mô phỏng thí nghiệm của em với các làn đường có thông số như sau:

Làn đường Thông số	1	2	3	4	5	6	7	8
Độ rộng(m)	20	25	15	30	20	30	20	30
Khoảng cách tới làn tiếp theo(m)	60	95	35	60	100	50	110	N/A
Hằng số tỉ lệ λ ($\frac{\text{điểm}}{m^2}$)	0.008	0.0002	0.0005	0.003	0.0004	0.007	0.009	0.0002

Cụ thể với các tham số của hàm chi phí là $C = \sqrt[3]{C_{cat} \cdot C_{st} \cdot C_{bw}} = 0.2$ và $c_{ij} = 0.05 * rand(0,1)$

Với trường hợp này kết quả thuật toán PSO_Based thường tốt hơn khoảng 15% so với thuật toán PSO.

Tuy nhiên thuật toán PSO có tính ổn định không cao, thường xuyên rơi vào trường hợp tối mà không thể thoát ra được tối ưu cục bộ. Do đó để làm nổi bật phần mô phỏng này em lấy một trường hợp mà thuật toán PSO chạy tệ.

Sinh dữ liệu và lấy khung trên 1km chiều dài.

Em thu được kết quả như sau:

Kết quả Tham số	Thuật toán PSO	Thuật toán PSO_Based
Độ bao phủ	75.8%	48,6%
Chi phí	1.4365	0.4102
Số điểm được gửi lên server	593 điểm	236 điểm

Tuy thuật toán PSO cho độ bao phủ tốt hơn nhưng cả 2 yếu tố là giá trị hàm chi phí và số điểm được gửi lên server đều lớn hơn rất nhiều so với thuật toán PSO_Based do em đề xuất. Hình biểu diễn vùng bao phủ có màu xanh, đường được tô bởi màu đỏ. Cụ thể như hình dưới đây:

