

Điều khiển tránh va chạm cho robot e-puck sử dụng kiến trúc phản ứng với mạng nơ-ron nhân tạo trong Webots

Tóm tắt

Bài viết này trình bày một phương pháp điều khiển phản ứng sử dụng mạng nơ-ron nhân tạo (ANN) cho robot e-puck trong môi trường mô phỏng Webots nhằm thực hiện nhiệm vụ tránh va chạm. Mô hình điều khiển sử dụng dữ liệu cảm biến khoảng cách để ra quyết định về chuyển động của robot dựa trên luật IF-THEN kết hợp trọng số hành vi được thiết kế mô phỏng mạng nơ-ron đơn giản. Kết quả thử nghiệm cho thấy robot có thể phản ứng nhanh và chính xác với các chướng ngại vật, minh chứng cho tính hiệu quả và đơn giản của kiến trúc điều khiển phản xạ này.

1. Giới thiệu

Trong điều khiển robot di động, tránh va chạm là một trong những hành vi cơ bản nhưng quan trọng, đặc biệt trong môi trường không xác định hoặc có nhiều vật cản. Các kiến trúc điều khiển phản ứng, điển hình như kiến trúc của Brooks và mô hình Braitenberg, đã được áp dụng rộng rãi do sự đơn giản và hiệu quả của chúng. Bài viết này đề xuất một mô hình điều khiển dựa trên mạng nơ-ron nhân tạo đơn giản với đầu vào là cảm biến khoảng cách và đầu ra là vận tốc bánh xe, phù hợp với triết lý điều khiển phản ứng tức thời.

2. Mô hình và phương pháp

2.1. Kiến trúc phản ứng với mạng nơ-ron

Kiến trúc phản ứng được xây dựng dựa trên mạng nơ-ron nhân tạo đơn lớp, với hàm truyền là tổ hợp tuyến tính các cảm biến đầu vào có trọng số:

$$y = f\left(\sum_i w_i x_i\right)$$

Trong đó x_i là tín hiệu từ cảm biến khoảng cách, w_i là trọng số điều chỉnh cho mỗi cảm biến, và y là vận tốc của động cơ trái/phải.

2.2. Thiết lập mô phỏng trong Webots

- **Robot:** e-puck, 2 bánh xe và 8 cảm biến khoảng cách (ps0 đến ps7)
- **Môi trường:** có chứa các vật cản ngẫu nhiên
- **Timestep:** sử dụng thời gian mặc định từ Webots

2.3. Thuật toán tránh vật cản

Dựa trên phân vùng cảm biến:

- Trái: ps5, ps6, ps7
- Phải: ps0, ps1, ps2
- Trước: ps3, ps4

Robot sẽ:

- **Đi thẳng** nếu không có vật cản
- **Rẽ phải/trái** nếu phát hiện vật cản bên trái/phải
- **Lùi lại** nếu có vật cản phía trước và hai bên

Chi tiết mã nguồn được trình bày trong đoạn mã Python sử dụng API của Webots.

3. Kết quả thực nghiệm

Robot đã được kiểm tra trong các kịch bản khác nhau (có vật cản trước, trái, phải, hoặc tổ hợp). Kết quả ghi nhận:

- Robot không xảy ra va chạm

- Phản ứng nhanh với các chướng ngại vật
- Di chuyển ổn định trong môi trường giả lập

Một video minh họa thí nghiệm đã được ghi lại và có thể được truy cập để đánh giá trực quan hiệu năng của thuật toán.

4. Thảo luận

Phương pháp phản ứng sử dụng mạng nơ-ron đơn giản tỏ ra phù hợp với bài toán tránh vật cản cơ bản. Tuy nhiên, mô hình chưa có cơ chế học hay thích nghi, nên hiệu suất có thể suy giảm trong môi trường động. Việc mở rộng mô hình sang mạng nhiều lớp hoặc học trọng số tự động có thể là hướng phát triển tiếp theo.

5. Kết luận

Bài viết đã trình bày một giải pháp điều khiển tránh va chạm cho robot e-puck dựa trên kiến trúc phản ứng kết hợp mạng nơ-ron nhân tạo đơn giản. Mô hình đã được triển khai trong Webots và cho thấy hiệu quả trong việc xử lý tình huống tránh vật cản mà không cần lập kế hoạch hành vi phức tạp.

Tài liệu tham khảo

1. Brooks, R. (1986). A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*.
2. Arkin, R. (1987). Motor schema based navigation for a mobile robot.
3. Phạm Duy Hưng, *Slide bài giảng chương 2: Kiến trúc phản ứng*.
4. Bonabeau, E., Dorigo, M., & Theraulaz, G. (1999). *Swarm*

Intelligence: From Natural to Artificial Systems.

Đường dẫn mã nguồn và dữ liệu thống kê:
https://github.com/Quagn/Project_Distributed_Intelligent_Systems.git