

迷宮探索機器人

專題編號：102-CSIE-S018

執行期限：101 年第 1 學期至 102 年第 1 學期

指導教授：郭忠義

專題參與人員：99590308 張朕瑀

99590326 潘昆毅

一、摘要

本專題以 LEGO NXT 作為操縱之機器人，預先將擺設好的迷宮繪製於電腦上，電腦利用 A*演算法計算出最佳路徑，並藉由藍芽技術操控機器人走出迷宮。

關鍵詞：NXT、A*搜尋演算法、藍芽

二、緣由與目的

在機器人的世界裡，迷宮是個很困難的障礙，若是無厘頭的到處亂走，將無法順利走出迷宮。

藉由本專題使用之 A*搜尋演算法，不管在任意的迷宮中，甚至是自己隨意擺設的障礙下，只要有是有出口的迷宮，都能清楚的讓機器人走出最佳的路徑，順利的沿著結果並快速地離開迷宮。

三、系統架構與流程

(一) 系統架構

使用者利用視窗程式畫出複雜度不一的迷宮，利用 A*搜尋演算法找出該迷宮的最佳路徑後，經由藍芽傳輸路徑資料給機器人，再控制馬達行駛到目的地，在過程中如有突發障礙則停止，此設計如圖 1 所示。



圖 1. 系統設計概念

(二) 流程圖

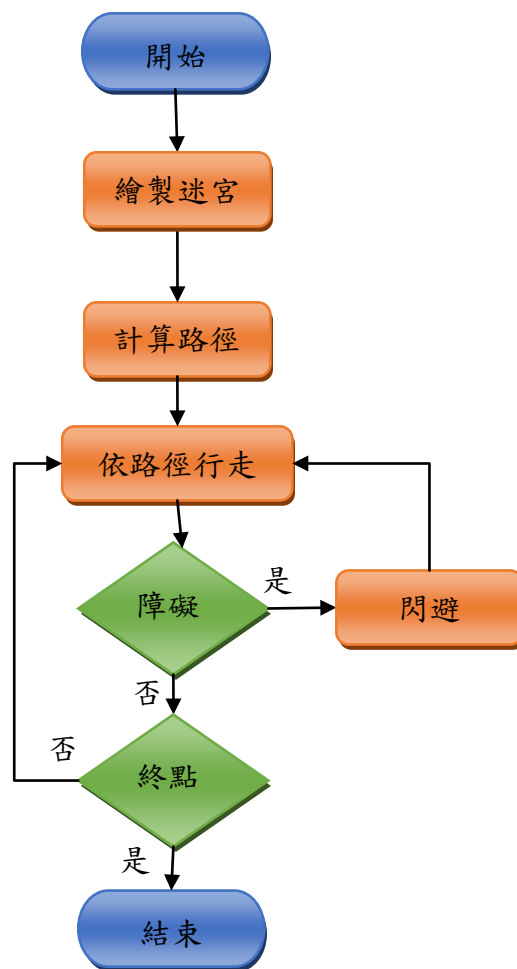


圖 2. 系統流程圖

四、實作重點

(一) 使用技術

1.NXT：

NXT 具有四個輸入端，與三個輸出端。輸入端包含：超音波偵測器、聲音偵測器、觸碰開關、光感測器。輸出端則可控制三部伺服馬達。

2. 藍芽：

藍芽是一種無線傳輸的技術，可以近距離作傳輸。NXT 機器人在藍芽傳輸上有兩種方法：一為信箱法 Mailbox，另一為指令法 Command。信箱法是指將要發送訊息放在正確的位置上，不經由 NXT 程式控制。指令法為將指令傳給 NXT，NXT 需要自行撰寫讀取通訊資料，在做出像對應的動作。而本專題使用信箱法。

3. A*搜尋演算法：

俗稱 A*演算法，是一種在圖形平面上，有多個節點的路徑，求出最低通過成本的演算法。常用於遊戲中的 NPC 的移動計算，或線上遊戲的 BOT 的移動計算上。該演算法像 Dijkstra 演算法一樣，可以找到一條最短路徑；也像 BFS 一樣，進行啟發式的搜尋。在此演算法中，如果以 $g(n)$ 表示從起點到任意頂點 n 的實際距離， $h(n)$ 表示任意頂點 n 到標的頂點的估算距離，那麼 A*演算法的公式為： $f(n)=g(n)+h(n)$ 。

的方式建構出迷宮，並能經由 A*演算法找出最佳路徑，我們認為本專題可以在幾何的方面有所幫助。

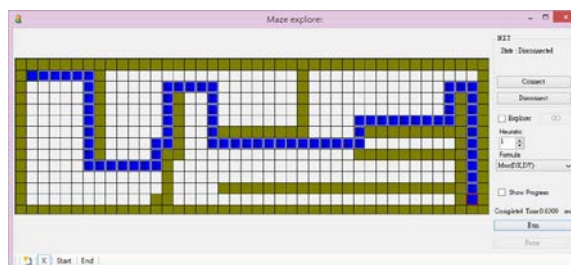


圖 3. 視窗程式



圖 4. 所運用之 NXT 機器人

(二) 環境背景

本專題利用 Visual Studio 2012 在 C# 語言環境下進行開發視窗程式，藉此呈現使用者環境畫面，再運用擴充 Library 傳達要求給 NXT 機器人進行相關功能，在 Windows 系統環境底下使用。

五、結論與應用

本專題在一開始本來要用傳統的老鼠走迷宮的方式來解迷宮，後來遇到一些困難，才想到要用 A*演算法來進行迷宮路徑的解法。實際運用後發現 A*演算法的啟發式搜尋很令人驚艷，而且還有許多不同的公式演算可以運用在不同的情況底下，對我們來說有很大的收穫。

除了路徑演算的問題外，我們在 NXT 機器人的行徑路線上遇到了會走偏、轉彎角度不一的問題，這方面的修正是我們一職努力的目標。

此專題開發可以讓使用者透過直觀

六、參考文獻

- [1] NXT
Lego, <http://mindstorms.lego.com/en-us/products/default.aspx>
- [2] A* Search Algorithm WIKI,
http://en.wikipedia.org/wiki/A*_search_algorithm
- [3] MindSqalls
<http://www.mindsqalls.net/Samples/Default.aspx>