

程式設計 (106-1)

期中專案

題目設計：孔令傑
國立臺灣大學資訊管理學系

說明：本專案題目的類似題目曾經出現在作業六和作業七。為了閱讀上的方便，在這份文件中我們完整地描述了整個問題。你可以利用你之前的想法或程式碼¹，不過除非你想多看一些範例，否則你不需要回去看那兩個題目的敘述。

1 題目敘述

當一架軍用飛機（戰鬥機、轟炸機、偵察機、補給機等等）要從 A 地飛往 B 地時，可能需要穿越敵方佈置的一些「威脅點」，例如雷達、防空砲、火箭基地等等。不同的威脅點有不同的威脅程度（攻擊力），例如被雷達掃到只是曝光而已，被火箭射中就直接墜毀了；不同的威脅點也有不同的威脅半徑（攻擊範圍）。在本專案中，我們將給定一個飛行任務，並請你在油料限制下，找出「成本」最小的飛行路線。以下我們將依序介紹「成本」的計算方式以及何謂油料限制。

1.1 單點風險

令 P_i 與 R_i 為威脅點 i 的威脅程度和威脅半徑，如果一架飛機和威脅點 i 的距離為 d ，則我們定義此飛機受到的來自此威脅點的「風險」為

$$P_i \max \left\{ \frac{R_i - d}{R_i}, 0 \right\},$$

亦即在威脅點正上方（距離為零）時會受到 100% 的風險、在威脅點的威脅半徑之外者零風險，而在威脅半徑內的風險則隨著距離拉遠而線性遞減。兩點間的距離以歐幾里得距離（Euclidean distance）計算，亦即若威脅點 i 的位置落在座標 (X_i, Y_i) 上，而飛機在 (x, y) ，單位皆為公里，則距離為 $d_i(x, y) = \sqrt{(X_i - x)^2 + (Y_i - y)^2}$ 公里。一架飛機受到的總風險為所有威脅點給他的風險的和，亦即若此飛機在 (x, y) ，共有 m 個威脅點，則其總風險為²

$$L(x, y) = \sum_{i=1}^m P_i \max \left\{ \frac{R_i - d_i(x, y)}{R_i}, 0 \right\}。$$

¹或許你應該感謝當時你有好好模組化，讓你的某些程式（特別是函數）可以被重新利用。如果你正在咒罵當時的你沒有好好模組化，也請不要介意，因為大部分人都是如此。從今天起好好模組化吧！

²如果你想看一些單點風險的計算範例，請自行複習作業六。

1.2 路段風險

會算單點的總風險後，我們要更進一步，來計算從一點到另一點的直線線段上的路段風險。直觀上來看，因為一個線段是由無限多個點集合起來的，我們似乎應該做個積分來算出路段風險，但因為這實在太麻煩了，所以我們簡化一點：我們假設飛行速率固定在每秒 1 公里，在路段上每隔 1 公里就標記一個點，然後把該路段上所有被標記的點（不含起點跟終點）的風險加總，來當作路段的風險。換言之，一個路段會被切成很多個小段，除了最後一小段以外，每段的長度都是 1 公里，而路段的風險就是這些小段的分段點的風險總和³。

1.3 路徑風險

會算路段風險後，我們要更進一步來算由起點到終點的路徑風險。一個路徑是由一或多個首尾相連的路段組成的，而兩個路段中間那個點被稱為轉折點。直觀上來看，一個路徑的風險就是其各路段之風險的總和，但實際上，由於路段的長度可能不是整數，因此如果第一條路段的最後一小段長度為 q ，則第二條路段的第一小段長度應該是 $1 - q$ ，亦即第二條路段的第一個要計算風險的點距離第一個轉折點的距離應該是 $1 - q$ ，依此類推。

舉例來說，如果威脅點 1、2、3 分別在 $(2, 5)$ 、 $(5, 4)$ 和 $(4, 2)$ ，威脅半徑依序為 2、3 和 2、威脅程度依序為 2、1 和 2，且我們要從起點 $s = (1, 1)$ 飛到終點 $t = (5, 5)$ 。以上資訊如圖 1 所示。

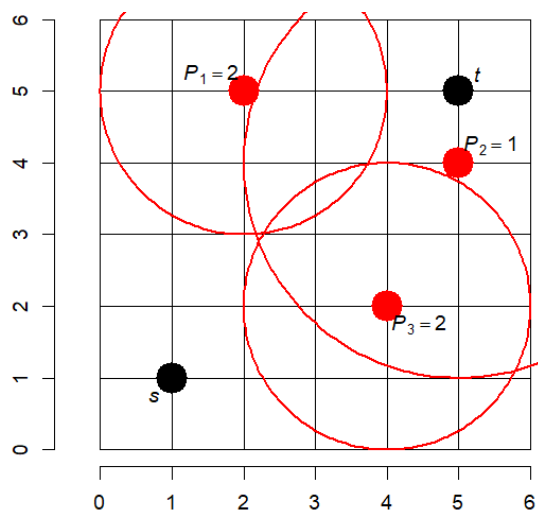


圖 1: 飛行任務範例

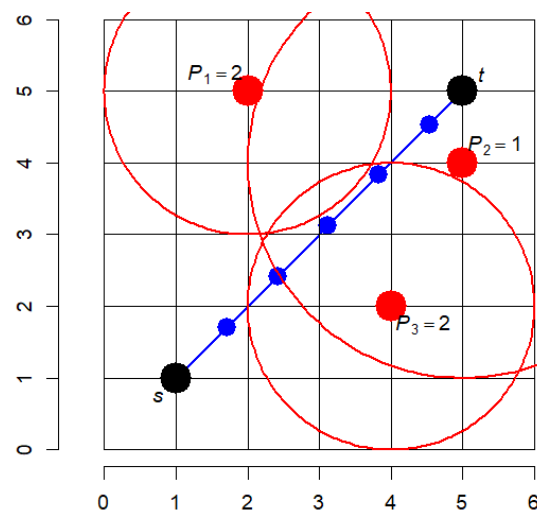


圖 2: 飛行任務範例：路徑一

我們來試算三條路徑的風險。首先，假設我們從 $(1, 1)$ 直接一條直線飛到 $(5, 5)$ 。

³如果你想看一些路段風險的計算範例，請自行複習作業七。

這條路徑只含有一個路段，長度為 $4\sqrt{2} \approx 5.657$ ，因此會有 5 個分段點（圖 2 中的藍色點），其中第 i 個點的座標可以被表示為

$$\left(1 + \frac{i}{\sqrt{2}}, 1 + \frac{i}{\sqrt{2}}\right),$$

亦即各點的座標為

$$(1.707, 1.707)、(2.414, 2.414)、(3.121, 3.121)、(3.828, 3.828)、(4.536, 4.536)。$$

這 5 點的風險分別約是 0、0.361、0.884、0.769 和 0.764，這條路段的風險約是 2.778，這也是這個路徑的風險。請注意起點和終點的風險是不含在內的。

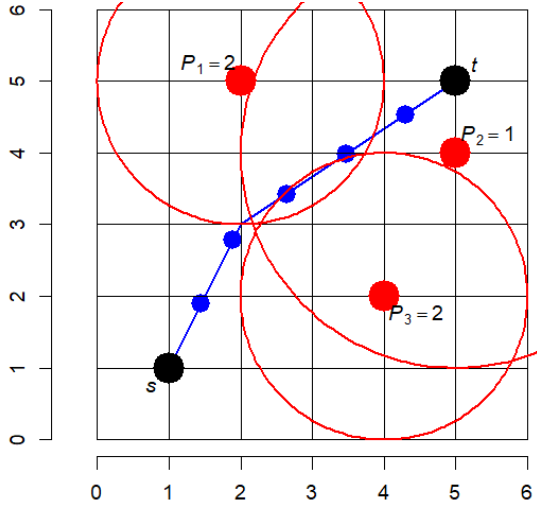


圖 3: 飛行任務範例：路徑二

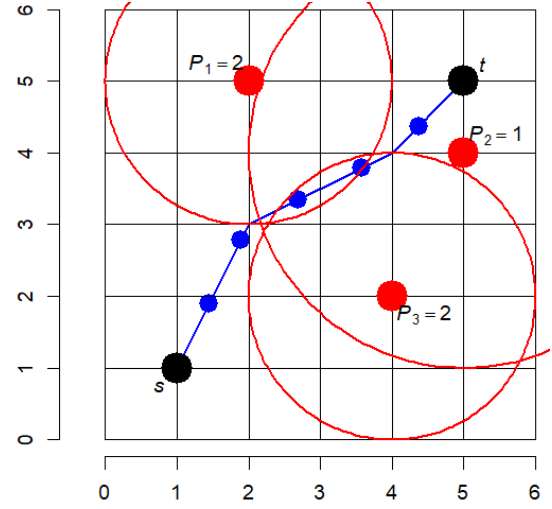


圖 4: 飛行任務範例：路徑三

我們的第二條路徑是從 (1,1) 先飛到 (2,3)，再飛到 (5,5)。這條路段長度為 $\sqrt{5} + \sqrt{13} \approx 5.842$ ，因此會有 5 個分段點（圖 3 中的藍色點），其中前兩點是

$$\left(1 + \frac{1}{\sqrt{5}}, 1 + \frac{2}{\sqrt{5}}\right) \approx (1.447, 1.894),$$

$$\left(1.447 + \frac{1}{\sqrt{5}}, 1.447 + \frac{2}{\sqrt{5}}\right) \approx (1.894, 2.789)。$$

此時距離轉折點 (2,3) 的距離只有 0.236，因此第三點跟轉折點的距離應該是 0.764，並且位在

$$\left(2 + (0.764)\left(\frac{3}{\sqrt{13}}\right), 3 + (0.764)\left(\frac{2}{\sqrt{13}}\right)\right) \approx (2.636, 3.424)。$$

最後兩點則位在

$$\left(2.636 + \frac{3}{\sqrt{13}}, 3.424 + \frac{2}{\sqrt{13}}\right) \approx (3.468, 3.978),$$

$$\left(3.468 + \frac{3}{\sqrt{13}}, 3.978 + \frac{2}{\sqrt{13}}\right) \approx (4.300, 4.533)。$$

這 5 點的風險分別約是 0、0、0.517、0.701 和 0.706，這條路徑的風險約是 1.925。繞個路確實可以讓風險降低。

我們的最後一條路徑，是從 (1, 1) 先飛到 (2, 3)，再飛到 (4, 4)，最後才飛到 (5, 5)。這條路段長度為 $\sqrt{5} + \sqrt{5} + \sqrt{2} \approx 5.886$ ，因此會有 5 個分段點（圖 4 中的藍色點），分別是

$$(1.447, 1.894)、(1.894, 2.789)、(2.683, 3.342)、(3.578, 3.789)、(4.373, 4.373)$$

請注意前兩點落在路段一、中間兩點落在路段二、最後一個點落在路段三。這 5 點的風險分別約是 0、0、0.523、0.694 和 0.756，這條路徑的風險約是 1.974。雖然多費了點心思繞路，但風險並沒有比路徑二低。

1.4 飛行成本

一個路徑的飛行成本被定義為路徑風險與轉折次數加權的和。更精確地說，假設一條路徑是由 k 個路段組成的，則這條路徑的成本為

$$\text{路徑風險} + w(k - 1),$$

其中 w 為給定的轉折權重，愈高表示轉折的成本愈高⁴。

在本專案中，我們想要找從起點 s 飛到終點 t 的最低成本路徑。以上一節的例子來說，顯然不論轉折權重為何，路徑三都比路徑二差，但路徑二跟路徑一誰好，就取決於轉折權重的大小了。如果轉折權重高於 $2.778 - 1.925 = 0.853$ ，則雖然路徑一風險較高，但加入轉折次數做考量後，就會變成路徑一比路徑二的成本更低了。

1.5 油料限制

本專案還有另一個議題，是飛行的油料限制。一架飛機在飛行中通常是無法加油的，我們也面對這個限制，因此我們也會被給定油料的上限。由於本題並不允許飛機改變速度（真是謝天謝地），因此相當於是給定飛行距離。任何路徑只要其總長度超過飛行距離上限，就是不可行（infeasible）的飛行計畫，反之即為可行（feasible）。舉例來說，如果飛行距離上限是 5.7，那麼上述的三條路徑中就只有路徑一是可行的了。

⁴至於為什麼飛機轉折會有「成本」，總之是飛行器研發單位說的，就是些轉彎飛比直接飛困難、操作上風險較高之類的原因，就不在本專案煩惱的範圍內了。

1.6 飛行路線最佳化

在本題中，你將被給定 m 個威脅點的座標、威脅半徑與威脅程度，以及一張橫座標與縱座標皆從 0、1、2 一直到 n 的地圖。你會另外被給定起點 s 和終點 t 的座標、轉折權重 w ，以及飛行距離上限 \bar{d} 。請在不違反油料限制的前提下，盡可能地找出成本最小的飛行路徑。

2 輸入輸出格式

系統會提供一共 25 組測試資料，每組測試資料裝在一個檔案裡。在每個檔案中，會有六列，第一列含有四個整數，依序是 n 、 m 、 w 和 \bar{d} 。第二列含有 m 個整數，第 i 個整數為第 i 個威脅點的 x 座標 X_i ，第三列含有 m 個整數，第 i 個整數為第 i 個威脅點的 y 座標 Y_i 。第四列含有 m 個整數，第 i 個整數為第 i 個威脅點的威脅半徑 R_i 。第五列含有 m 個整數，第 i 個整數為第 i 個威脅點的威脅程度 P_i 。第六列含有四個整數，依序是起點 s 的 x 座標 X_s 、起點 s 的 y 座標 Y_s 、終點 t 的 x 座標 X_t 、終點 t 的 y 座標 Y_t 。同一列的任兩個數字之間被一個空白字元隔開。已知 $n \in \{1, \dots, 1000\}$ 、 $m \in \{1, \dots, 10000\}$ 、 $w \in \{0, \dots, 1000\}$ 、 $\bar{d} \in \{n, \dots, 10n\}$ 、 $X_i \in \{0, \dots, n\}$ 、 $Y_i \in \{0, \dots, n\}$ 、 $X_s \in \{0, \dots, n\}$ 、 $Y_s \in \{0, \dots, n\}$ 、 $X_t \in \{0, \dots, n\}$ 、 $Y_t \in \{0, \dots, n\}$ 、 $R_i \in \{0, \dots, n\}$ 、 $P_i \in \{1, \dots, 1000\}$ ，且 $\bar{d} > \sqrt{(X_s - X_t)^2 + (Y_s - Y_t)^2}$ 。

讀入這些資料之後，請決定一條飛行路徑，並將轉折點依序印出。假設妳決定的路徑是從 (X_s, Y_s) 出發之後，依序在 (x_1, y_1) 、 (x_2, y_2) 一直到 (x_k, y_k) 做轉折，最後飛到 (X_t, Y_t) ，則請依序輸出 k 、 x_1 、 y_1 、 x_2 、 y_2 一直到 x_k 、 y_k 。兩個數字中間用一個空白字元隔開，最後面沒有空白。如果你想從 s 飛直線到 t ，就只要輸出 0 就好。舉例來說，如果輸入是

```
6 3 0 10
2 5 4
5 4 2
2 3 2
2 1 2
1 1 5 5
```

且你想要輸出上述的路徑一，則輸出應該是

```
0
```

PDOGS 會判定你的方案是可行的，成本則為 2.778。如果輸入是

```
6 3 1 10
2 5 4
5 4 2
2 3 2
2 1 2
1 1 5 5
```

且你想要輸出上述的路徑二，則輸出應該是

```
1 2 3
```

PDOGS 會判定你的方案是可行的，成本則為 $1.925 + 1 = 2.925$ 。如果你想要從 (1, 1) 走到 (1, 5) 再走到 (5, 1) 再走到 (5, 5)，則輸出應該是

```
2 1 5 5 1
```

PDOGS 會判定你的方案是不可行的。

你的.cpp 原始碼檔案裡面應該包含讀取測試資料、做運算，以及輸出答案的 C++ 程式碼。當然，你應該寫適當的註解。針對這個題目，你**可以**使用任何方法。

3 評分原則

這一題的其中 75 分會根據程式運算的結果給分。你的程式不需要找出真的能最小化總成本的最佳方案 (optimal solution)。只要你的輸出符合規定，且確實是一組可行方案 (feasible solution，在油料限制內由起點飛到終點)，就會得到分數。對於每一組輸入，PDOGS 會檢查你的輸出，如果輸出格式不合乎要求或方案不可行，則在該筆測試資料會得到零分；如果合乎要求，且目標式值非負，則對每筆測資，我們依下列公式計分：假設 z 是這組的路線成本、 z_1 是直接由起點飛直線到終點的成本、 z_0 是所有組的路線成本中最小的，則在這筆測資的得分就是

$$3 \left(\frac{z_1 - z}{z_1 - z_0} \right)。$$

以上面的例子而言，如果 $w = 0$ ，你找到的是路徑三，而其他組中最好的答案是路徑二，則你在這一筆測資的分數將是

$$3 \left(\frac{2.778 - 1.974}{2.778 - 1.925} \right) = 2.83。$$

寫程式之外，每組還需要合力用中文或英文寫一份書面報告（所謂「寫」，就是用電腦打的意思），以組為單位上傳 PDF 檔至 PDOGS。在報告裡請用文字描述你的演算法（可以用 pseudocode 但不能直接貼 code）、系統的設計（哪個函數做什麼、程式執行的流程等等）、分工方式（誰寫哪個函數、誰負責指揮、誰負責寫書面報告、誰負責買便當等等；當然一個人可以又買便當又寫程式），以及每個人的簡單心得感想。報告**不可以超過八面 A4 紙**。書面報告佔 25 分。這份專案截止後，書面報告才會被批改。

4 繳交方式

請修課的同學們自行組成每組三至四人的小組，以組為單位繳交你們的程式和報告。

有兩件事需要注意。首先，系統會以該組內任意一位同學的最後一次上傳得到的分數，做為該組所有人的分數，所以愈傳愈低分是有可能的。其次，原則上 PDOGS 不限制兩次上傳間的時間間隔，但如果有許多組在同個時間大量地上傳執行時間很長的程式，導致 PDOGS 大塞車，屆時我們會對兩次上傳的時間間隔做出限制。

程式的截止時間是 **2017 年 11 月 27 日凌晨一點**，書面報告的截止時間是 **2017 年 11 月 29 日凌晨一點**。