

2025年臺灣國際科學展覽會 優勝作品專輯

作品編號	010035
參展科別	數學
作品名稱	Japanese triangle之探討與推廣

就讀學校	高雄市立高雄高級中學 高雄市立高雄女子高級中學
指導教師	黃佩瑜 謝秉融
作者姓名	黃宇綸 黃宇瑄

關鍵詞	<u>三角形、迭代、高維</u>
-----	------------------

作者簡介



我是黃宇綸，目前就讀於高雄中學二年級。

從小我就對數學推理與科學研究有濃厚的興趣，尤其擅長數學與資訊領域。我曾多次參加數學科學展覽，這次的研究主題與計算機科學相關，對熱愛競程的我來說特別具有吸引力。研究的過程充滿挑戰，但我樂在其中，不僅鍛煉了我的邏輯思考與創造力，更展現了我自主學習與解決問題的能力。

我是黃宇瑄，目前就讀於高雄女中二年級電資班。

除了寫程式外，我最大的興趣是唱歌，目前是雄女合唱團的成員。這是我和雙胞胎弟弟第二次組隊參加國際科展，上一次是國二時。作為最有默契的研究夥伴，我們分工合作、共同討論，將一個個想法轉化為實際作品，過程既充實又令人感到無比成就感。

2025 年臺灣國際科學展覽會

研究報告

區別：

科別：數學科

作品名稱：*Japanese triangle* 之探討與推廣

關 鍵 詞：三角形、迭代、高維

編號：

目 錄

摘要	01
壹、前言	02
一、研究動機	02
二、研究目的	02
三、文獻回顧	02
四、研究工具	02
貳、研究過程與方法	02
研究一： $\Delta_{n,\ell}$ 之最佳路徑最小化問題	04
研究二： $\Delta_{n,\ell}$ 之最糟路徑最大化問題	12
研究三： $\Delta^*_{n,\ell}$ 之最佳路徑最小化問題	18
研究四： $\Delta^*_{n,\ell}$ 之最糟路徑最大化問題	28
研究五： $\ell=1$ 高維空間情形之探討	38
參、研究結果	46
肆、討論與未來展望	48
伍、參考文獻資料	48
附錄：程式碼	

Japanese triangle 之探討與推廣

Research on Japanese triangle

摘要

本作品在探討 2023 年 IMO 問題 5 中所提到的關於日式三角形(*Japanese triangle*)之問題，日式三角形是將 $1+2+\cdots+n$ 個圓排成正三角形的形狀，使得對所有 $i=1,2,\cdots,n$ ，由上往下數的第 i 列有 i 個圓，且每一列都有一個圓塗成紅色。日式三角形中的忍者路徑是一串由最上列到最下列的 n 個圓，其中每個圓連到其下一列與之相鄰的兩圓之一。我們分成兩個研究方向：一、找出 k 的最大值，保證在每一個日式三角形中，有一條包含至少 k 個紅色圓的忍者路徑。二、找出 k 的最小值，保證在每一個日式三角形中，有一條包含至多 k 個紅色圓的忍者路徑。研究中，我們一般化每列的紅圓數為任意自然數 ℓ (若該列總圓數不足 ℓ 則以該列總圓數塗色)，並將問題推廣至空間三角垛的情形。最後，我們將 $\ell=1$ 的情形推廣至高維空間。

Abstract

The paper explores the problem concerning the *Japanese triangle* mentioned in Problem 5 of the 2023 IMO. Let n be a positive integer. A *Japanese triangle* consists of $1+2+\cdots+n$ circles arranged in an equilateral triangular shape such that for each $i=1,2,\cdots,n$, the i^{th} row contains exactly i circles, exactly one of which is coloured red. A *ninja path* in a Japanese triangle is a sequence of n circles obtained by starting in the top row, then repeatedly going from a circle to one of the two circles immediately below it and finishing in the bottom row. Our research is divided into two directions:

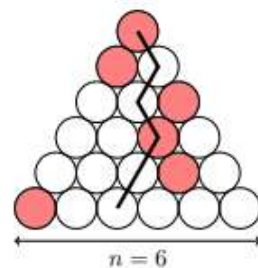
1. In terms of n , find the greatest k such that in each Japanese triangle there is a ninja path containing at least k red circles.
2. In terms of n , find the smallest k such that in each Japanese triangle there is a ninja path containing at most k red circles.

In our research, we generalize the number of red circles in each column to be any natural number ℓ (if the total number of circles in the column is less than ℓ , the color will be colored based on the total number of circles in the column), and extend the problem to the case of a spatial triangular stack. Finally, we extend the case of $\ell=1$ to high-dimensional spaces.

壹、前言

一、研究動機

2023 年 IMO 問題 5：「設 n 為正整數。日式三角形(Japanese triangle)是將 $1+2+\cdots+n$ 個圓排成正三角形的形狀，使得對所有 $i=1,2,\cdots,n$ ，由上往下數的第 i 列有 i 個圓，且每一列都有一個圓塗成紅色。在日式三角形中，所謂的忍者路徑(ninja path)，是一串由最上列到最下列的 n 個圓，其中每個圓連到其下一列與之相鄰的兩圓之一。試找出 k 的最大值，保證在每一個日式三角形中，有一條包含至少 k 個紅色圓的忍者路徑。」



若將每一列都有一個圓塗成紅色改成每一列都有 ℓ 個圓塗成紅色(若該列總圓數不足 ℓ 則以該列總圓數塗色)，或是將平面三角形改成空間三角垛排列，結果又是如何呢？我們展開一連串的研究。

二、研究目的

- (一)找出 k 的最大值，保證在每一個日式三角形中，有一條包含至少 k 個紅色圓的忍者路徑。
- (二)找出 k 的最小值，保證在每一個日式三角形中，有一條包含至多 k 個紅色圓的忍者路徑。
- (三)一般化每一列的紅圓數為任意自然數 ℓ 。
- (四)將平面日式三角形的情形，推廣到空間日式三角垛的情形。
- (五)將 $\ell=1$ 的情形推廣至高維空間。

三、文獻回顧

我們參考了網路上外國網友發表的各種題解，採用與文獻[2]類似的解題方式更嚴謹的證明了 IMO 原題，並加以推廣。

四、研究工具

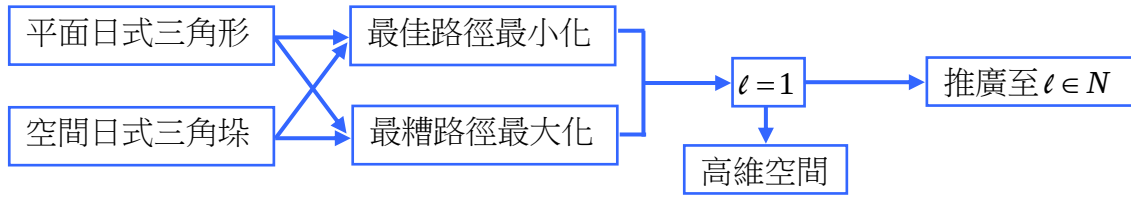
筆、電腦、C++、Python、GeoGebra

貳、研究過程與方法

題意說明

原題中「找出 k 的最大值，保證在每一個日式三角形中，有一條包含至少 k 個紅色圓的忍者路徑。」，意同於「每一個 n 階日式三角形中，均有一條由最上列到最下列通過最多紅圓的最佳路徑，求這些最佳路徑所通過的紅圓數的最小值 k 。」我們將這個求 k 值的過程簡稱「最佳路徑最小化」。同理，「找出 k 的最小值，保證在每一個日式三角形中，有一條包含至多 k 個紅色圓的忍者路徑。」之求 k 值的過程簡稱「最糟路徑最大化」。

研究流程



名詞與符號定義 (研究一)(推廣時所使用的名詞與符號補充定義於各研究內容前)

- (一) n 階 ℓ 型日式三角形(記作 $\Delta_{n,\ell}$)：將 $1+2+\cdots+n$ 個圓排成正三角形的形狀，使得對所有 $i=1,2,\cdots,n$ ，由上往下數的第 i 列有 i 個圓，且第 i 列塗成紅色的圓數為 $\min\{i,\ell\}$ 。
- (二)忍者路徑：一條路徑通過日式三角形中最上列到最下列的圓，其中每個圓連到其下一列與之相鄰的兩圓之一。如圖 1，藍色路徑即為一條通過 4 個紅圓的忍者路徑。
- (三)最佳路徑：通過最多紅圓數的忍者路徑。如圖 1，藍色路徑即為此日式三角形由最上列至最下列的最佳路徑(不一定唯一)。
- (四)最糟日式三角形：由最上列到最下列，最佳路徑所通過的紅圓數為最小的日式三角形。
- (五) $J_\ell(n)$ ： n 階 ℓ 型最糟日式三角形中，由最上列到最下列，其最佳路徑所通過的紅圓數。
- (六) (i,j) ：日式三角形中，由上往下數的第 i 列中，由左往右數的第 j 個圓之坐標。
- (七) $f(i,j)$ ：由 $(1,1)$ 至 (i,j) 的最佳路徑所通過的紅圓數。如圖 2， $f(5,3)=3$ 。
- (八)第 i 列之最佳序列： $(f(i,1), f(i,2), \cdots, f(i,i))$ 。如圖 3，第 5 列之最佳序列為 $(3,3,3,2,2)$ 。
- (九)最佳序列之次狀態列：給定一最佳序列 $(f(i,1), f(i,2), \cdots, f(i,i))$ ，則其次狀態列為

$$(f(i,1), \text{Max}\{f(i,1), f(i,2)\}, \text{Max}\{f(i,2), f(i,3)\}, \cdots, \text{Max}\{f(i,i-1), f(i,i)\}, f(i,i)),$$

記作 $(f(i,1), f(i,2), \cdots, f(i,i))'$ 。例如： $(4,5,2,3,1)' = (4,5,5,3,1)$ 。

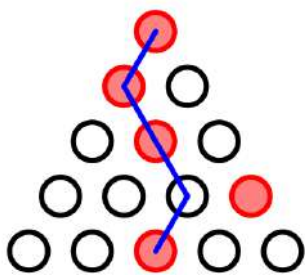


圖 1

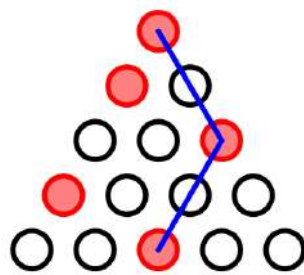


圖 2

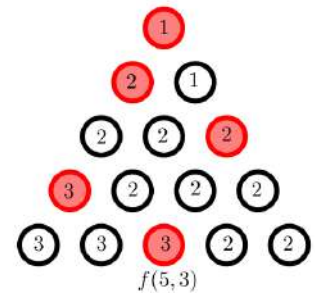
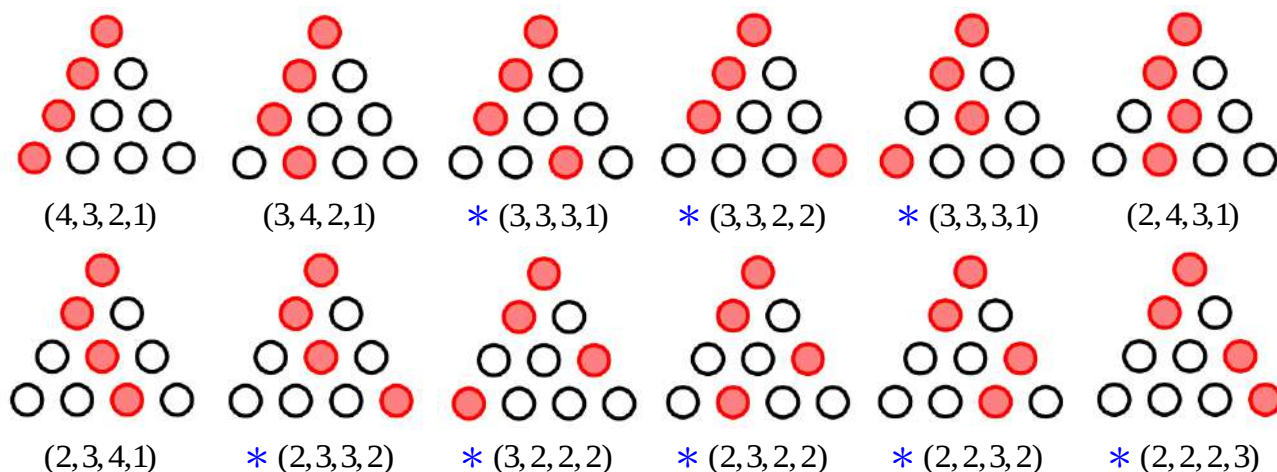


圖 3

【例】窮舉法列出所有的 $\Delta_{4,1}$ 以及其第 4 列之最佳序列。(共 24 個，因對稱性僅需列出 12 個。)



由上可知，標示藍星的日式三角形即為最糟日式三角形，且 $J_1(4) = 3$ 。

研究一： $\Delta_{n,\ell}$ 之最佳路徑最小化問題

首先，我們將 $\Delta_{n,\ell}$ 中 $f(i, j)$ 的求值規則整理於下：

$\Delta_{n,\ell}$ 中 $f(i, j)$ 的求值規則：

$$f(i, j) = \begin{cases} f(i-1, j) + c & , j = 1 \\ \text{Max}\{f(i-1, j-1), f(i-1, j)\} + c, & 2 \leq j \leq i-1, \text{其中當 } (i, j) \text{ 為紅圓時 } c = 1, \text{反之 } c = 0. \\ f(i-1, j-1) + c & , j = i \end{cases}$$

由以上求值規則可得到以下性質：

性質 1：在 $\Delta_{n,\ell}$ 中，第 i 列最佳序列總和 = 第 $(i-1)$ 列最佳序列之次狀態列總和 + ℓ 。

(一) $\ell = 1$ (IMO 原題)

首先，我們針對較小的 n 值做嘗試，找出 $J_1(n)$ 值，整理於下表：

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$J_1(n)$	1	2	2	3	3	3	3	4	4	4	4	4	4	4	4	5

我們可以輕易的觀察出規律，並得到 $J_1(n) = \lceil \log_2 n \rceil + 1$ 的

結果。在研究的過程中，我們也發現，要建構出最糟日式三角形並不難，它的排列原則為每列的紅圓盡量錯開上一列紅圓下方相鄰的兩圓。如圖 4 的紅圓排列(若第 i 列紅圓座標為 (i, j) ，則第 $i+1$ 列紅圓座標為 $(i+1, j')$ ，其中 $j+2 \equiv j' \pmod{i+1}, 1 \leq j' \leq i+1$)，便可得出其中一個最糟 $\Delta_{7,1}$ ，其中第 7 列的最佳序列為 $(3, 3, 3, 3, 3, 3, 3)$ ，我們將這樣的排列法稱為「斜列錯開法」。

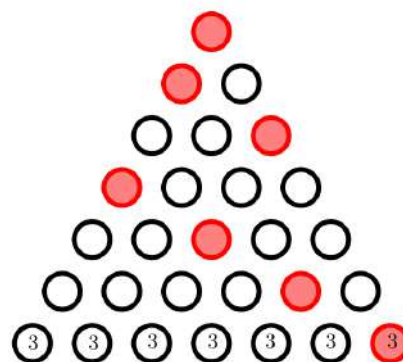


圖 4

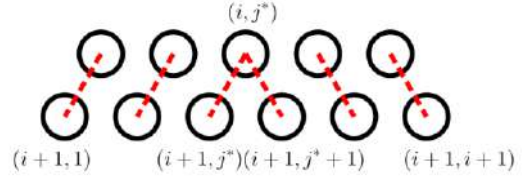
接下來，我們要證明觀察出來的公式是正確的。(參考自文獻[2])

定理 2：在 $\Delta_{n,1}$ 中， $J_1(n) = \lceil \log_2 n \rceil + 1$ 。

【證明】

令 $S_i = \sum_{j=1}^i f(i, j)$ ，即第 i 列之最佳序列總和。

Claim1. $S_{i+1} \geq 1 + S_i + \left\lceil \frac{S_i}{i} \right\rceil, \forall i \in N$



Proof. 令第 i 列之最佳序列中最大值為 $f(i, j^*)$ ，顯然 $f(i, j^*) \geq \left\lceil \frac{S_i}{i} \right\rceil$ 。

$$\begin{aligned} \text{則 } S_{i+1} &= \sum_{j=1}^{i+1} f(i+1, j) = \sum_{j=1}^{j^*} f(i+1, j) + \sum_{j=j^*+1}^{i+1} f(i+1, j) \\ &\geq 1 + \sum_{j=1}^{j^*} f(i, j) + \sum_{j=j^*}^i f(i, j) = 1 + S_i + f(i, j^*) \geq 1 + S_i + \left\lceil \frac{S_i}{i} \right\rceil \end{aligned}$$

Claim2. 若 $n = 2^{k-1} + r, 0 \leq r \leq 2^{k-1} - 1$ ，則 $S_n \geq n(k-1) + 2r + 1, \forall n \in N$ 。

Proof. 當 $n = 1 = 2^0 + 0$ ，即 $k = 1, r = 0$ ， $S_1 = 1 \geq 1 \cdot (1-1) + 2 \cdot 0 + 1$ 原式成立

設 $n = 2^{k-1} + r, 0 \leq r \leq 2^{k-1} - 1$ 時，原式成立，即 $S_n \geq n(k-1) + 2r + 1$ ，

$$\text{則 } \left\lceil \frac{S_n}{n} \right\rceil \geq \left\lceil \frac{n(k-1) + 2r + 1}{n} \right\rceil = \left\lceil (k-1) + \frac{2r+1}{n} \right\rceil = k$$

$$(\because 0 \leq r \leq 2^{k-1} - 1 \therefore 1 \leq 2r + 1 = (r+1) + r \leq 2^{k-1} + r = n \Rightarrow 0 < \frac{2r+1}{n} \leq 1)$$

分成 2 種 case 考慮 S_{n+1} ：

(1) 當 $0 \leq r \leq 2^{k-1} - 2$ 時， $n+1 = 2^{k-1} + (r+1)$

$$S_{n+1} \geq 1 + S_n + \left\lceil \frac{S_n}{n} \right\rceil \geq 1 + n(k-1) + 2r + 1 + k = (n+1)(k-1) + 2(r+1) + 1$$

(2) 當 $r = 2^{k-1} - 1$ 時， $n+1 = 2^k + 0$

$$S_{n+1} \geq 1 + S_n + \left\lceil \frac{S_n}{n} \right\rceil \geq 1 + n(k-1) + 2r + 1 + k = 1 + nk - n + 2^k - 2 + 1 + k = (n+1)k + 1$$

(1)(2) 皆成立 由數學歸納法得證

由 Claim2 知， $f(n, j^*) \geq \left\lceil \frac{S_n}{n} \right\rceil \geq \left\lceil \frac{n(k-1) + 2r + 1}{n} \right\rceil = \left\lceil (k-1) + \frac{2r+1}{n} \right\rceil = k$ ，而我們確實可以

「斜列錯開法」建構出一個 $\Delta_{n,1}$ 滿足 $f(n, j^*) = k$ ，故 $J_1(n) = k = \log_2(n-r) + 1 = \lceil \log_2 n \rceil + 1$ 。 ■

【 $\lceil \log_2 n \rceil + 1$ 的涵義】 n 可以執行幾次「除以 2」的運算後小於 1，即 n 以二進位表示的位數。

(二) $\ell = 2$

我們同樣先針對較小的 n 值做嘗試，找出 $J_2(n)$ 值，整理於下表：

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$J_2(n)$	1	2	3	4	4	5	5	5	6	6	6	6	6	7	7	7
n	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
$J_2(n)$	7	7	7	7	8	8	8	8	8	8	8	8	8	8	9	9

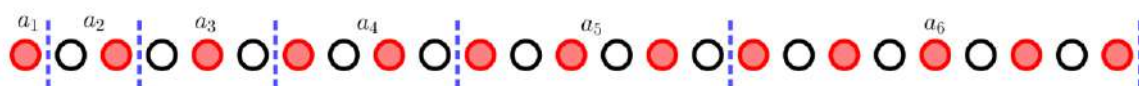
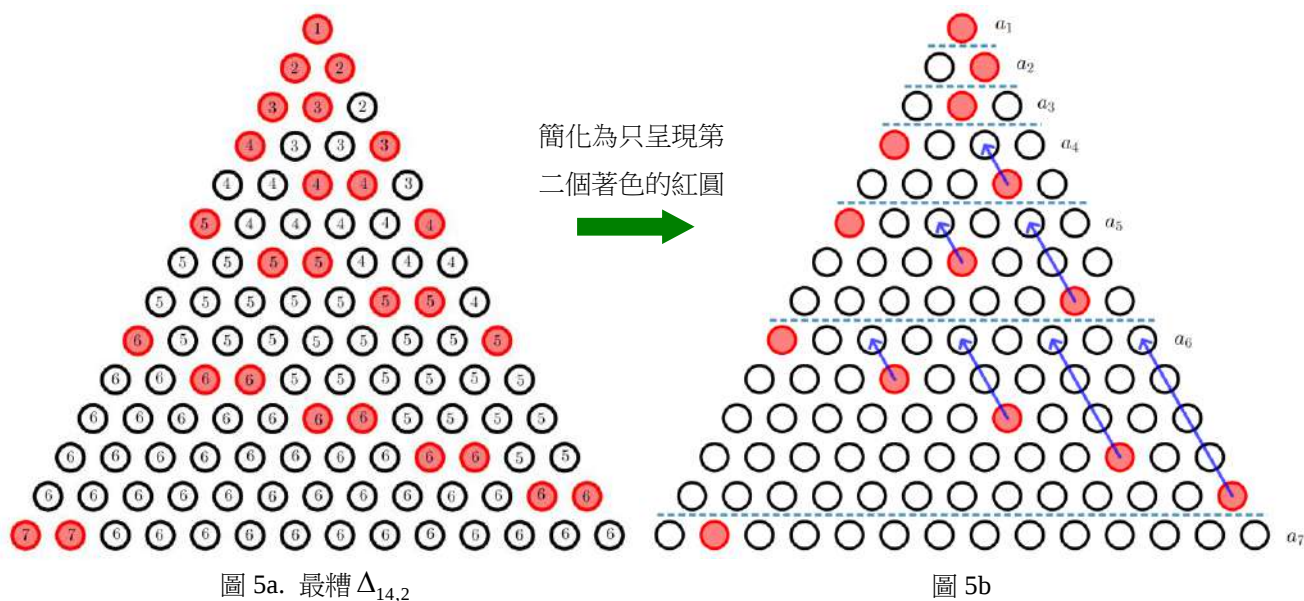
在觀察 $J_2(n)$ 的公式時並不顯然，所以我們鎖定滿足 $J_2(n) = J_2(n-1) + 1$ 的 n 值來作研究。

設 a_k 為滿足 $J_2(n) = k$ 的最小 n 值，可得數列 $\langle a_k \rangle$: 1, 2, 3, 4, 6, 9, 14, 21, 31, ……，而這個數列會

$$\text{滿足遞迴關係式} \begin{cases} a_k = 1 + \left\lceil \frac{\sum_{i=1}^{k-1} a_i}{2} \right\rceil, k \geq 2 \\ a_1 = 1 \end{cases}.$$

問題：為什麼會有這個遞迴關係式呢？

分析：我們以圖 5a~5c 的例子來說明。圖 5a 為我們以「斜列錯開法」建構出的最糟 $\Delta_{14,2}$ ，其排列原則為每一列的紅圓位置盡量錯開上一列紅圓下方相鄰的兩圓，即若第 i 列著色的圓依序為 $(i, j_1), (i, j_2)$ ，則第 $i+1$ 列著色的圓依序為 $(i+1, j_1'), (i+1, j_2')$ ，其中 $j_2 + 2 \equiv j_1' \pmod{i+1}$ ， $j_2 + 3 \equiv j_2' \pmod{i+1}$ ， $1 \leq j_1', j_2' \leq i+1$ 。事實上，其前 i 列 ($i = 1, 2, \dots, 14$) 所形成的三角形皆為最糟 $\Delta_{i,2}$ 。為了研究方便，我們只記錄每一列第二個著色的紅圓位置(圖 5b)，並可觀察到第 $a_k \sim (a_{k+1} - 1)$ 列的最佳序列之最大值均為 k 。將圖 5b 中第 $a_1, a_2, a_3, a_4, a_5, a_6$ 列取出排成一列(圖 5c)，並如圖 5b 所示補上紅圓，可知整列圓共有 $\left\lceil \frac{a_1 + a_2 + \dots + a_6}{2} \right\rceil$ 個紅圓，又一行恰一個紅圓，即代表第 a_7 列前共有 $\left\lceil \frac{a_1 + a_2 + \dots + a_6}{2} \right\rceil$ 列，因此 $a_7 = 1 + \left\lceil \frac{a_1 + a_2 + \dots + a_6}{2} \right\rceil$ 。

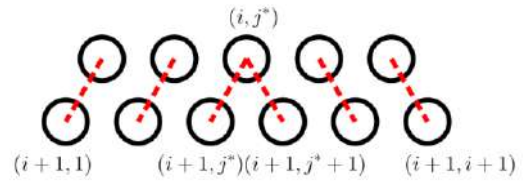


不過，這畢竟只是一個特例，我們仍然得嚴謹的證明 $J_2(n)$ 與 $\langle a_k \rangle$ 確實是具有以上關係，且我們也想要找出 $\langle a_k \rangle$ 的通式。然而我們查詢 OEIS，雖然可以找到相同數列 [A005428](#)，但完全無法找到這個數列的通式，顯然要表示這個通式是有困難的，因此我們採用另一個方式來描述 $\Delta_{n,2}$ 最佳路徑最小化的情形。

定理 3：在 $\Delta_{n,2}$ 中，令數列 $\langle a_k \rangle$ 滿足 $\begin{cases} a_k = 1 + \left\lfloor \frac{\sum_{i=1}^{k-1} a_i}{2} \right\rfloor, k \geq 2 \\ a_1 = 1 \end{cases}$ ，若 $a_k \leq n < a_{k+1}$ ，則 $J_2(n) = k$ 。

【證明】

令 $S_i = \sum_{j=1}^i f(i, j)$ ，即第 i 列之最佳序列總和。



Claim1. $S_{i+1} \geq 2 + S_i + \left\lfloor \frac{S_i}{i} \right\rfloor, \forall i \in N$

Proof. 令第 i 列之最佳序列中最大值為 $f(i, j^*)$ ，顯然 $f(i, j^*) \geq \left\lfloor \frac{S_i}{i} \right\rfloor$ 。

$$\begin{aligned} \text{則 } S_{i+1} &= \sum_{j=1}^{i+1} f(i+1, j) = \sum_{j=1}^{j^*} f(i+1, j) + \sum_{j=j^*+1}^{i+1} f(i+1, j) \\ &\geq 2 + \sum_{j=1}^{j^*} f(i, j) + \sum_{j=j^*}^i f(i, j) = 2 + S_i + f(i, j^*) \geq 2 + S_i + \left\lfloor \frac{S_i}{i} \right\rfloor \end{aligned}$$

Claim2. 若 $n = a_k + r, 0 \leq r \leq a_{k+1} - a_k - 1, k \geq 2$ ，則

(1) 當 $a_1 + a_2 + \cdots + a_{k-1}$ 為奇數時， $S_n \geq n(k-1) + 3r + 2$ 。

(2) 當 $a_1 + a_2 + \cdots + a_{k-1}$ 為偶數時， $S_n \geq n(k-1) + 3r + 1$ 。

Proof. 當 $n = 2 = a_2 + 0$ ，即 $k = 2, r = 0$ ， $a_1 = 1$ 為奇數， $S_2 = 4 \geq 2 \cdot (2-1) + 3 \cdot 0 + 2$ 原式成立

設 $n = a_k + r, 0 \leq r \leq a_{k+1} - a_k - 1$ 時，原式成立 即

(1) 當 $a_1 + a_2 + \cdots + a_{k-1}$ 為奇數時， $S_n \geq n(k-1) + 3r + 2$ ，

$$\text{此時 } a_{k+1} - a_k = \left(1 + \left\lfloor \frac{a_1 + a_2 + \cdots + a_k}{2} \right\rfloor \right) - \left(1 + \left\lfloor \frac{a_1 + a_2 + \cdots + a_{k-1}}{2} \right\rfloor \right) = \left\lfloor \frac{a_k}{2} \right\rfloor$$

上式原因如下：

$$\textcircled{1} a_k \text{ 為奇數時, } a_{k+1} - a_k = \frac{a_1 + a_2 + \cdots + a_k}{2} - \frac{a_1 + a_2 + \cdots + a_{k-1} + 1}{2} = \frac{a_k - 1}{2} = \left\lfloor \frac{a_k}{2} \right\rfloor$$

$$\textcircled{2} a_k \text{ 為偶數時, } a_{k+1} - a_k = \frac{a_1 + a_2 + \cdots + a_k + 1}{2} - \frac{a_1 + a_2 + \cdots + a_{k-1} + 1}{2} = \frac{a_k}{2} = \left\lfloor \frac{a_k}{2} \right\rfloor$$

$$\text{則 } \left\lceil \frac{S_n}{n} \right\rceil \geq \left\lceil \frac{n(k-1)+3r+2}{n} \right\rceil = \left\lceil (k-1) + \frac{3r+2}{n} \right\rceil = k$$

$$(\because 0 \leq r \leq \left\lfloor \frac{a_k}{2} \right\rfloor - 1 \therefore 2 \leq 3r+2 = 2r+2+r \leq 2 \cdot \left\lfloor \frac{a_k}{2} \right\rfloor + r \leq 2 \cdot \frac{a_k}{2} + r = n \Rightarrow 0 < \frac{3r+2}{n} \leq 1)$$

分成 2 種 case 考慮 S_{n+1} :

$$(a) \text{當 } 0 \leq r \leq a_{k+1} - a_k - 2 = \left\lfloor \frac{a_k}{2} \right\rfloor - 2 \text{ 時, } n+1 = a_k + (r+1),$$

$$\text{則 } S_{n+1} \geq 2 + S_n + \left\lceil \frac{S_n}{n} \right\rceil \geq 2 + n(k-1) + 3r+2 + k = (n+1)(k-1) + 3(r+1) + 2$$

$$(b) \text{當 } r = a_{k+1} - a_k - 1 = \left\lfloor \frac{a_k}{2} \right\rfloor - 1 \text{ 時, } n+1 = a_{k+1} + 0$$

$$\begin{aligned} \text{則 } S_{n+1} &\geq 2 + S_n + \left\lceil \frac{S_n}{n} \right\rceil \geq 2 + n(k-1) + 3r+2 + k = (n+1)k + 3r + 4 - n \\ &= (n+1)k + (2r+4) - a_k = (n+1)k + 2 \left\lfloor \frac{a_k}{2} \right\rfloor + 2 - a_k \end{aligned}$$

$$\textcircled{1} a_k \text{ 為奇數時, } S_{n+1} \geq (n+1)k + 2 \cdot \frac{a_k-1}{2} + 2 - a_k = (n+1)k + 3 \cdot 0 + 1$$

$$\textcircled{2} a_k \text{ 為偶數時, } S_{n+1} \geq (n+1)k + 2 \cdot \frac{a_k}{2} + 2 - a_k = (n+1)k + 3 \cdot 0 + 2$$

(a)(b)皆成立 由數學歸納法得證

(2)同理(1)之證明過程, 我們可以證得, 當 $a_1 + a_2 + \cdots + a_{k-1}$ 為偶數時,

$$S_n \geq n(k-1) + 3r + 1. \text{ (完整證明於研究日誌)}$$

由 Claim2 知, 若 $n = a_k + r, 0 \leq r \leq a_{k+1} - a_k - 1, k \geq 2$, 當 $a_1 + a_2 + \cdots + a_{k-1}$ 為奇數時,

$$f(n, j^*) \geq \left\lceil \frac{S_n}{n} \right\rceil \geq \left\lceil \frac{n(k-1)+3r+2}{n} \right\rceil = \left\lceil (k-1) + \frac{3r+2}{n} \right\rceil = k. \text{ 同理當 } a_1 + a_2 + \cdots + a_{k-1} \text{ 為偶數}$$

$$\text{時, } f(n, j^*) \geq \left\lceil \frac{S_n}{n} \right\rceil \geq \left\lceil \frac{n(k-1)+3r+1}{n} \right\rceil = \left\lceil (k-1) + \frac{3r+1}{n} \right\rceil = k. \text{ 而我們確實可以「斜列錯開}$$

法」建構出一個 $\Delta_{n,2}$ 滿足 $f(n, j^*) = k$, 故若 $a_k \leq n < a_{k+1}$, 則 $J_2(n) = k$. ■

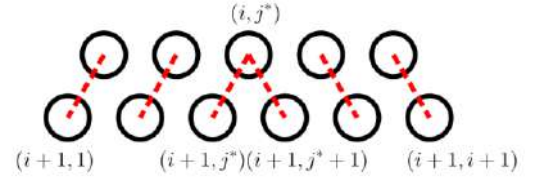
(三) $\ell \in \mathbb{N}$

接下來, 我們要探討日式三角形一般化的情形, 也就是 ℓ 為任意自然數的情形。

定理 4: 在 $\Delta_{n,\ell}$ 中, 令數列 $\langle a_k \rangle$ 滿足 $\begin{cases} a_k = 1 + \left\lfloor \frac{\sum_{i=1}^{k-1} a_i + \frac{(\ell-1)(\ell-2)}{2}}{\ell} \right\rfloor, k \geq \ell+1 \\ a_1 = 1, a_2 = 2, \dots, a_\ell = \ell \end{cases}, \text{ 若 } a_k \leq n < a_{k+1},$ 則 $J_\ell(n) = k$。
--

【證明】

令 $S_i = \sum_{j=1}^i f(i, j)$ ，即第 i 列之最佳序列總和。



Claim1. $S_i = i^2$, $\forall 1 \leq i \leq \ell$ 且 $S_{i+1} \geq \ell + S_i + \left\lceil \frac{S_i}{i} \right\rceil$, $\forall i \geq \ell$

Proof. 顯然 $S_i = i^2$, $\forall 1 \leq i \leq \ell$ 是成立的，我們只需證明不等式中 $i \geq \ell$ 的情形。

令第 i 列之最佳序列中最大值為 $f(i, j^*)$ ，顯然 $f(i, j^*) \geq \left\lceil \frac{S_i}{i} \right\rceil$ 。

$$\begin{aligned} \text{則 } S_{i+1} &= \sum_{j=1}^{i+1} f(i+1, j) = \sum_{j=1}^{j^*} f(i+1, j) + \sum_{j=j^*+1}^{i+1} f(i+1, j) \\ &\geq \ell + \sum_{j=1}^{j^*} f(i, j) + \sum_{j=j^*}^i f(i, j) = \ell + S_i + f(i, j^*) \geq \ell + S_i + \left\lceil \frac{S_i}{i} \right\rceil \end{aligned}$$

Claim2. 若 $n = a_k + r$, $0 \leq r \leq a_{k+1} - a_k - 1$, $k \geq \ell$ ，則

當 $a_1 + a_2 + \dots + a_{k-1} + \frac{(\ell-1)(\ell-2)}{2} \equiv t \pmod{\ell}$, $t = 1, 2, \dots, \ell$, $k \geq 2$ 時，

$$S_n \geq n(k-1) + (\ell+1)r + (\ell+1-t)。$$

Proof. 當 $n = \ell = a_\ell + 0$ ，即 $k = \ell, r = 0$

$$a_1 + a_2 + \dots + a_{\ell-1} + \frac{(\ell-1)(\ell-2)}{2} = 1 + 2 + \dots + (\ell-1) + \frac{(\ell-1)(\ell-2)}{2} = (\ell-1)^2 \equiv 1 \pmod{\ell}$$

$$S_\ell = \ell^2 \geq \ell(\ell-1) + (\ell+1) \cdot 0 + (\ell+1-1) = \ell^2 \quad \text{原式成立}$$

設 $n = a_k + r$, $0 \leq r \leq a_{k+1} - a_k - 1$, $k \geq \ell$ 時，原式成立 即

$$\text{當 } a_1 + a_2 + \dots + a_{k-1} + \frac{(\ell-1)(\ell-2)}{2} \equiv t \pmod{\ell} \text{ 時， } S_n \geq n(k-1) + (\ell+1)r + (\ell+1-t)，$$

此時

$$\begin{aligned} a_{k+1} - a_k &= \left(1 + \left\lceil \frac{a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2}}{\ell} \right\rceil \right) - \left(1 + \left\lceil \frac{a_1 + a_2 + \dots + a_{k-1} + \frac{(\ell-1)(\ell-2)}{2}}{\ell} \right\rceil \right) \\ &= \left\lceil \frac{a_k + (t-1)}{\ell} \right\rceil \end{aligned}$$

$$\text{則 } \left\lceil \frac{S_n}{n} \right\rceil \geq \left\lceil \frac{n(k-1) + (\ell+1)r + (\ell+1-t)}{n} \right\rceil = \left\lceil (k-1) + \frac{(\ell+1)r + (\ell+1-t)}{n} \right\rceil = k$$

$$\text{上式原因如下： } \because 0 \leq r \leq \left\lceil \frac{a_k + (t-1)}{\ell} \right\rceil - 1$$

$$\therefore \ell+1-t \leq (\ell+1)r + (\ell+1-t) = \ell \cdot r + (\ell+1-t) + r \leq \ell \cdot \left(\left\lceil \frac{a_k + (t-1)}{\ell} \right\rceil - 1 \right) + (\ell+1-t) + r$$

$$\leq \ell \cdot \frac{a_k + (t-1)}{\ell} + 1 - t + r = a_k + r = n \Rightarrow 0 < \frac{(\ell+1)r + (\ell+1-t)}{n} \leq 1$$

分成 2 種 case 考慮 S_{n+1} :

(a) 當 $0 \leq r \leq a_{k+1} - a_k - 2 = \left\lfloor \frac{a_k + (t-1)}{\ell} \right\rfloor - 2$ 時, $n+1 = a_k + (r+1)$,

$$\begin{aligned} \text{則 } S_{n+1} &\geq \ell + S_n + \left\lfloor \frac{S_n}{n} \right\rfloor \geq \ell + n(k-1) + (\ell+1)r + (\ell+1-t) + k \\ &= (n+1)(k-1) + (\ell+1)(r+1) + (\ell+1-t) \end{aligned}$$

(b) 當 $r = a_{k+1} - a_k - 1 = \left\lfloor \frac{a_k + (t-1)}{\ell} \right\rfloor - 1$ 時, $n+1 = a_{k+1} + 0$

$$\begin{aligned} \text{則 } S_{n+1} &\geq \ell + S_n + \left\lfloor \frac{S_n}{n} \right\rfloor \geq \ell + n(k-1) + (\ell+1)r + (\ell+1-t) + k \\ &= (n+1)k + (\ell+1)r + (2\ell+1-t) - n = (n+1)k + \ell \cdot r + (2\ell+1-t) - a_k \\ &= (n+1)k + \ell \left\lfloor \frac{a_k + (t-1)}{\ell} \right\rfloor + (\ell+1-t) - a_k \end{aligned}$$

令 $a_k \equiv s \pmod{\ell}$, $s = 1, 2, \dots, \ell$, 即 $a_k = \ell \cdot p + s$, $p \in N$,

$$\text{則 } a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2} \equiv t + s \pmod{\ell}$$

① 若 $t+s \leq \ell$, 則 $a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2} \equiv t + s \pmod{\ell}$

$$\text{且 } \left\lfloor \frac{a_k + (t-1)}{\ell} \right\rfloor = p = \frac{a_k - s}{\ell} \quad (\because a_k + (t-1) = \ell \cdot p + s + (t-1) \leq \ell \cdot p + (\ell-1))$$

$$\begin{aligned} \text{故 } S_{n+1} &\geq (n+1)k + \ell \left\lfloor \frac{a_k + (t-1)}{\ell} \right\rfloor + (\ell+1-t) - a_k \\ &= (n+1)k + \ell \left(\frac{a_k - s}{\ell} \right) + (\ell+1-t) - a_k = (n+1)k + (\ell+1) \cdot 0 + [\ell+1-(t+s)] \end{aligned}$$

② 若 $t+s > \ell$, 則 $a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2} \equiv t + s - \ell \pmod{\ell}$

$$\text{且 } \left\lfloor \frac{a_k + (t-1)}{\ell} \right\rfloor = p+1 = \frac{a_k - s}{\ell} + 1 = \frac{a_k - s + \ell}{\ell}$$

$$(\because a_k + (t-1) = \ell \cdot p + s + (t-1) > \ell \cdot p + (\ell-1) = \ell(p+1) - 1)$$

$$\begin{aligned} \text{故 } S_{n+1} &\geq (n+1)k + \ell \left\lfloor \frac{a_k + (t-1)}{\ell} \right\rfloor + (\ell+1-t) - a_k \\ &= (n+1)k + \ell \left(\frac{a_k - s + \ell}{\ell} \right) + (\ell+1-t) - a_k = (n+1)k + (\ell+1) \cdot 0 + [\ell+1-(t+s-\ell)] \end{aligned}$$

(a)(b)皆成立 由數學歸納法得證

由 Claim1、Claim2 知, 若 $1 \leq n < \ell$, $f(n, j^*) \geq \left\lfloor \frac{S_n}{n} \right\rfloor = \left\lfloor \frac{n^2}{n} \right\rfloor = n = a_n$

若 $n = a_k + r$, $0 \leq r \leq a_{k+1} - a_k - 1$, $k \geq \ell$,

$$f(n, j^*) \geq \left\lceil \frac{S_n}{n} \right\rceil \geq \left\lceil \frac{n(k-1) + (\ell+1)r + (\ell+1-t)}{n} \right\rceil = \left\lceil (k-1) + \frac{(\ell+1)r + (\ell+1-t)}{n} \right\rceil = k。$$

而我們確實可以「斜列錯開法」建構出一個 $\Delta_{n,\ell}$ 滿足 $f(n, j^*) = k$ ，因此，

若 $a_k \leq n < a_{k+1}$ ，則 $J_\ell(n) = k$ 。■

【直觀證明】

以上我們使用代數的方式證明了定理 4，然而證明過程複雜且冗長，因此我們開始思考，是否有一種直觀的方法可以證明 $J_\ell(n)$ 與數列 $\langle a_k \rangle$ 的關係，也就是說，若我們能說明「斜列錯開法」確實可建構出最糟日式三角形(不會再更糟了)，那麼只需針對這個排列法去求 $J_\ell(n)$ 的通式即可。以下描述「平面斜列錯開法」的排列規則：

平面斜列錯開法：

在 $\Delta_{n,\ell}$ 中，設第 i 列紅圓坐標為 (i, j_s) , $1 \leq s \leq m_i$ ，其中 $m_i = \min\{i, \ell\}$ ，則第 $i+1$ 列的紅圓坐標為 $(i+1, j'_s)$, $1 \leq s \leq m_{i+1}$ ，其中 $j_{m_i} + s + 1 \equiv j'_s \pmod{i+1}$, $1 \leq j'_s \leq i+1$ 。

我們觀察定理 4 證明中 Claim1 的部分，它說明了 S_{i+1} 與 S_i 的關係。

問題 1：不等式 $S_{i+1} \geq \ell + S_i + \left\lceil \frac{S_i}{i} \right\rceil$ 等號何時成立？

分析：觀察其推導過程，在 S_i 已知的狀況下，

$$S_{i+1} = \sum_{j=1}^{i+1} f(i+1, j) = \sum_{j=1}^{j^*} f(i+1, j) + \sum_{j=j^*+1}^{i+1} f(i+1, j)$$

$$\geq \ell + \sum_{j=1}^{j^*} f(i, j) + \sum_{j=j^*}^i f(i, j) = \ell + S_i + f(i, j^*) \geq \ell + S_i + \left\lceil \frac{S_i}{i} \right\rceil$$

由上式可知，當 $\begin{cases} f(i+1, j) = f(i, j) + c, & 1 \leq j \leq j^* \\ f(i+1, j) = f(i, j-1) + c, & j^*+1 \leq j \leq i+1 \end{cases}$ ，其中 $(i+1, j)$ 為紅圓時 $c=1$ ，反

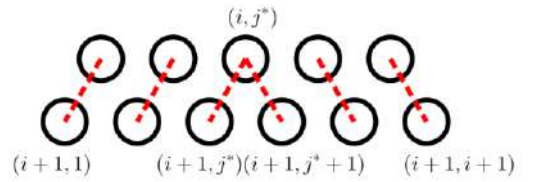
之 $c=0$ ，且 $f(i, j^*) = \left\lceil \frac{S_i}{i} \right\rceil$ 時，等號便會成立，此時 S_{i+1} 會達到最小值，而第 $i+1$ 列之最佳

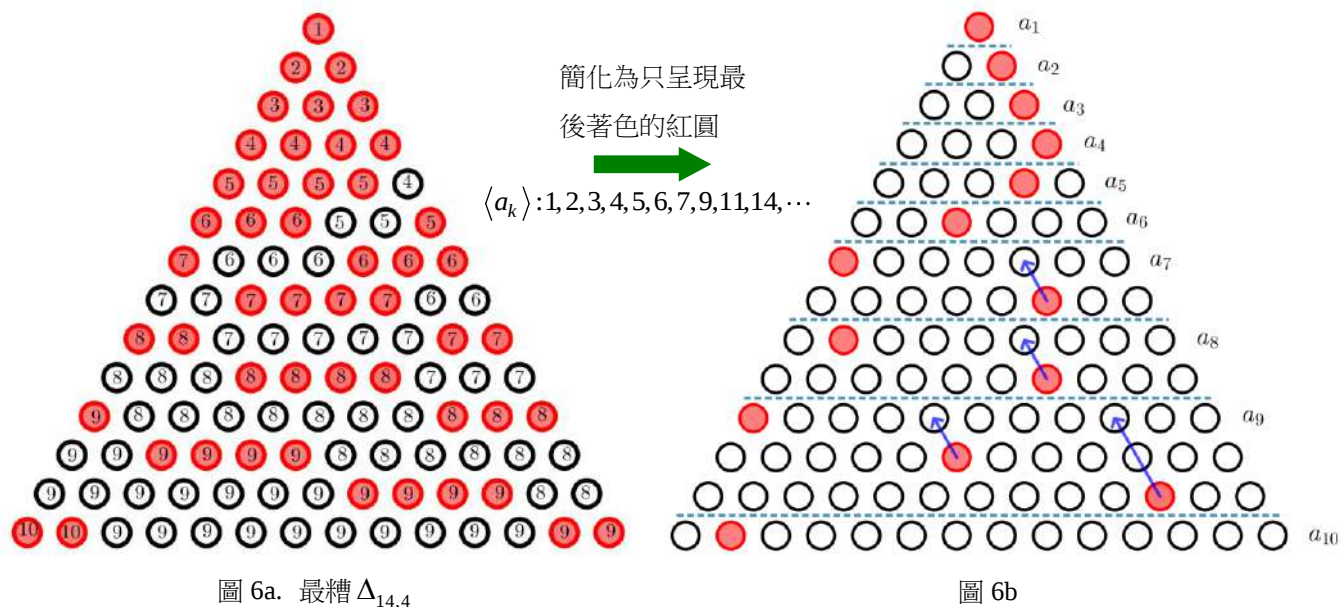
序列為只由 $\left\lceil \frac{S_{i+1}}{i+1} \right\rceil$ 與 $\left(\left\lceil \frac{S_{i+1}}{i+1} \right\rceil - 1 \right)$ 組成的分布時，其由第 1 列至第 $i+1$ 列的最佳路徑所通過

的紅圓數為 $\left\lceil \frac{S_{i+1}}{i+1} \right\rceil$ ，且不可能再少了！依此類推，便可建構出任何大小的最糟日式三角形。

顯然「斜列錯開法」所建構出的 $\Delta_{n,\ell}$ 可滿足這個目標，因此我們只需要針對這樣的特例去

尋找其每列的最佳序列最大值，便可得知 $J_\ell(n)$ 的值，事實上這個值就是 $\left\lceil \frac{S_n}{n} \right\rceil$ 。





問題 2：定理 4 中的數列 $\langle a_k \rangle$ 之遞迴關係式如何得知？

分析：以圖 6a 做說明，它是一個以「斜列錯開法」建構出的最糟 $\Delta_{14,4}$ ，其第 $a_k \sim (a_{k+1}-1)$ 列的最佳序列之最大值均為 k 。將圖 6b 中第 $a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9$ 列取出排成一列於圖 6c，並如圖 6b 所示補上紅圓，同時第 a_2, a_3 列須再分別補上 2 個與 1 個空圓，

因此整列圓中共有 $\left\lceil \frac{a_1 + \dots + a_9 + (1+2)}{4} \right\rceil$ 個紅圓，又一行恰一個紅圓，即第 a_{10} 列前共有

$\left\lceil \frac{a_1 + \dots + a_9 + (1+2)}{4} \right\rceil$ 列，因此 $a_{10} = 1 + \left\lceil \frac{a_1 + \dots + a_9 + (1+2)}{4} \right\rceil$ 。同理在最糟 $\Delta_{n,\ell}$ 中，取出

的第 $a_2 \sim a_{\ell-1}$ 列的圓必須要補上 $1+2+\dots+(\ell-2)$ 個，因此 $a_k = 1 + \left\lceil \frac{\sum_{i=1}^{k-1} a_i + \frac{(\ell-1)(\ell-2)}{2}}{\ell} \right\rceil$ 。

由以上討論，可證明「平面斜列錯開法」確實可建構出最糟日式三角形並得到定理 4 的結果。

研究二： $\Delta_{n,\ell}$ 之最糟路徑最大化問題

解決了 $\Delta_{n,\ell}$ 之最佳路徑最小化問題後，我們將問題改成：「找出 k 的最小值，保證在每一個日式三角形中，有一條包含至多 k 個紅色圓的忍者路徑。」為了研究需要，補充幾個名詞與符號定義於下：

名詞與符號定義 (研究二)

(一)最糟路徑：通過最少紅圓數的忍者路徑。

(二)最佳日式三角形：由最上列到最下列，最糟路徑所通過的紅圓數為最大的日式三角形。

(三) $I_\ell(n)$ ： n 階 ℓ 型最佳日式三角形中，由最上列到最下列，其最糟路徑所通過的紅圓數。

(四) $g(i, j)$ ：由 $(1,1)$ 至 (i, j) 的最糟路徑所通過的紅圓數。

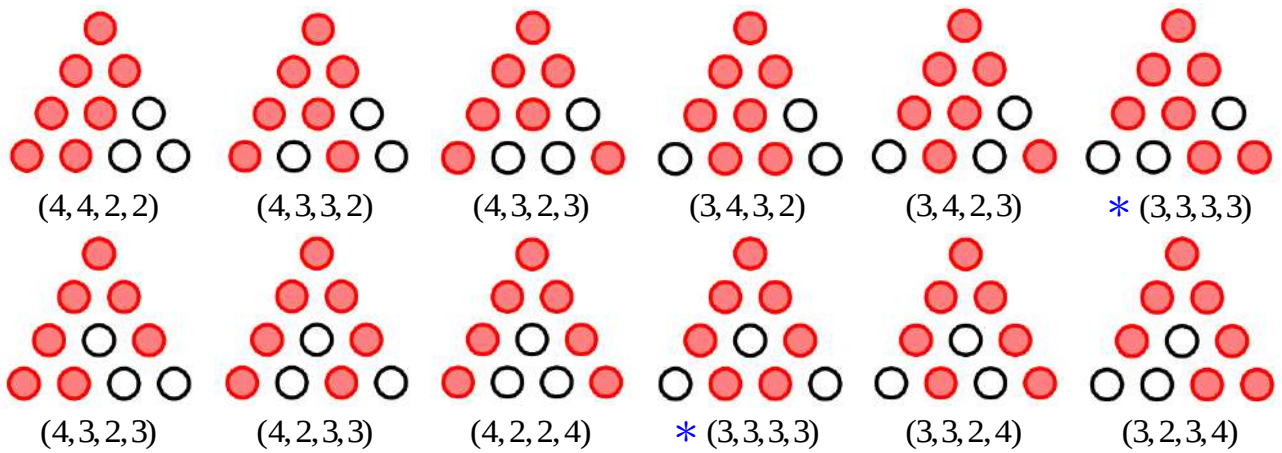
(五)第 i 列之最糟序列： $(g(i,1), g(i,2), \dots, g(i,i))$ 。

(六)最糟序列之次狀態列：給定一最糟序列 $(g(i,1), g(i,2), \dots, g(i,i))$ ，則其次狀態列為

$$(g(i,1), \min\{g(i,1), g(i,2)\}, \min\{g(i,2), g(i,3)\}, \dots, \min\{g(i, i-1), g(i,i)\}, g(i,i)) ,$$

記作 $(g(i,1), g(i,2), \dots, g(i,i))'$ 。例如： $(4, 5, 2, 3, 1)' = (4, 4, 2, 2, 1, 1)$ 。

【例】窮舉法列出所有的 $\Delta_{4,2}$ 以及其第 4 列之最糟序列。(共 18 個，因對稱性僅需列出 12 個。)



由上可知，標示藍星的日式三角形即為最佳日式三角形，且 $I_2(4) = 3$ 。

同研究一，我們將 $\Delta_{n,\ell}$ 中 $g(i, j)$ 的求值規則整理於下：

$\Delta_{n,\ell}$ 中 $g(i, j)$ 的求值規則：

$$g(i, j) = \begin{cases} g(i-1, j) + c & , j = 1 \\ \min\{g(i-1, j-1), g(i-1, j)\} + c, & 2 \leq j \leq i-1, \text{其中當 } (i, j) \text{ 為紅圓時 } c = 1, \text{反之 } c = 0. \\ g(i-1, j-1) + c & , j = i \end{cases}$$

由以上求值規則可得到以下性質：

性質 5：在 $\Delta_{n,\ell}$ 中，第 i 列最糟序列總和 = 第 $(i-1)$ 列最糟序列之次狀態列總和 + ℓ 。

(一) $\ell = 1$

當 $\ell = 1$ 時， $I_1(n) = 1$ 是顯然成立的，因為每個紅圓下方均有兩個圓與之相臨且至少有一個空圓，除了第一列起始紅圓外，必存在一條忍者路徑不再通過其它紅圓，也就是說，任意排列的 $\Delta_{n,1}$ 皆是最佳日式三角形。而建構最佳日式三角形的原則是盡可能在空圓下方安排紅圓，避免路徑一路經過太多空圓，依據這個原則，我們創造出「斜列防堵法」。

比如圖 7，便是利用「斜列防堵法」建構的 7 階最佳日式三角形，其第 7 列之最糟序列為 $(2, 2, 2, 2, 2, 2, 1)$ ，可以觀察到，縱使我們很想中斷任何一條從第二列開始就沒通過紅圓的路徑，但這是辦不到的，因為每列空圓增加 1 個，單用 1 顆紅圓不足以阻擋第二列起全空圓路徑的發生，因此 $I_1(n)=1, \forall n \in N$ 。我們將這個結論寫成定理 6。

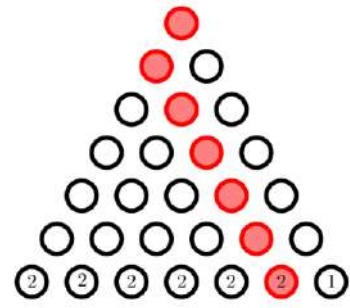


圖 7

定理 6：在 $\Delta_{n,1}$ 中， $I_1(n)=1$ 。

(二) $\ell=2$

我們先針對較小的 n 值做嘗試，找出 $I_2(n)$ 值，整理於下表：

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$J_1(n)$	1	2	2	3	3	3	3	4	4	4	4	4	4	4	4	5

由上表可以很輕易的觀察出規律，並得到 $I_2(n)=\lceil \log_2 n \rceil +1$ 的結果，這竟然和 $J_1(n)$ 的公式是相同的。我們同樣可以「斜列防堵法」建構出最佳日式三角形，圖 8 便是其中一個 7 階最佳日式三角形，其中第 7 列之最糟序列為 $(4, 4, 4, 4, 4, 4, 3)$ 。

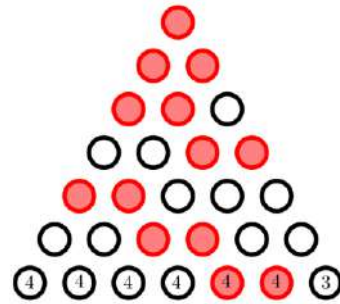


圖 8

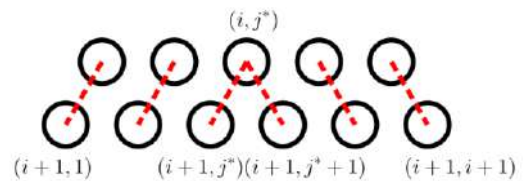
接下來，我們要來證明觀察出來的公式是正確的。

定理 7：在 $\Delta_{n,2}$ 中， $I_2(n)=\lceil \log_2 n \rceil +1$ 。

【證明】

令 $S_i = \sum_{j=1}^i g(i, j)$ ，即第 i 列之最糟序列總和。

Claim1. $S_{i+1} \leq 2 + S_i + \left\lceil \frac{S_i}{i} \right\rceil, \forall i \in N$



Proof. 令第 i 列之最糟序列中最小值為 $g(i, j^*)$ ，顯然 $g(i, j^*) \leq \left\lceil \frac{S_i}{i} \right\rceil$ 。

$$\begin{aligned}
 \text{則 } S_{i+1} &= \sum_{j=1}^{i+1} g(i+1, j) = \sum_{j=1}^{j^*} g(i+1, j) + \sum_{j=j^*+1}^{i+1} g(i+1, j) \\
 &\leq 2 + \sum_{j=1}^{j^*} g(i, j) + \sum_{j=j^*}^i g(i, j) = 2 + S_i + g(i, j^*) \leq 2 + S_i + \left\lceil \frac{S_i}{i} \right\rceil
 \end{aligned}$$

Claim2. 若 $n = 2^{k-1} + r, 0 \leq r \leq 2^{k-1} - 1$ ，則 $S_n \leq nk + 2r, \forall n \in N$ 。

Proof. 當 $n = 1 = 2^0 + 0$ ，即 $k = 1, r = 0$ ， $S_1 = 1 \leq 1 \cdot 1 + 2 \cdot 0$ 原式成立

設 $n = 2^{k-1} + r, 0 \leq r \leq 2^{k-1} - 1$ 時，原式成立，即 $S_n \leq nk + 2r$ ，

$$\text{則 } \left\lfloor \frac{S_n}{n} \right\rfloor \leq \left\lfloor \frac{nk + 2r}{n} \right\rfloor = \left\lfloor k + \frac{2r}{n} \right\rfloor = k$$

$$(\because 0 \leq r \leq 2^{k-1} - 1 \therefore 0 \leq 2r = r + r \leq 2^{k-1} - 1 + r = n - 1 \Rightarrow 0 \leq \frac{2r}{n} < 1)$$

分成 2 種 case 考慮 S_{n+1} ：

(1) 當 $0 \leq r \leq 2^{k-1} - 2$ 時， $n + 1 = 2^{k-1} + (r + 1)$

$$S_{n+1} \leq 2 + S_n + \left\lfloor \frac{S_n}{n} \right\rfloor \leq 2 + nk + 2r + k = (n + 1)k + 2(r + 1)$$

(2) 當 $r = 2^{k-1} - 1$ 時， $n + 1 = 2^k + 0$

$$S_{n+1} \leq 2 + S_n + \left\lfloor \frac{S_n}{n} \right\rfloor \leq 2 + nk + 2r + k = 2 + nk + 2^k - 2 + k = (n + 1)(k + 1)$$

(1)(2)皆成立 由數學歸納法得證

由 Claim2 知， $g(n, j^*) \leq \left\lfloor \frac{S_n}{n} \right\rfloor \leq \left\lfloor \frac{nk + 2r}{n} \right\rfloor = \left\lfloor k + \frac{2r}{n} \right\rfloor = k$ ，而我們確實可以「斜列防堵法」

建構出一個 $\Delta_{n,2}$ 滿足 $g(n, j^*) = k$ ，故 $I_2(n) = k = \log_2(n - r) + 1 = \lceil \log_2 n \rceil + 1$ 。 ■

(三) $\ell \in N$

在 $\ell = 2$ 的研究中，意外發現 $I_2(n) = J_1(n)$ ，那麼 $I_\ell(n) = J_{\ell-1}(n)$ 會成立嗎？可惜的是，答案是否定的，這引起我們的好奇心， $I_\ell(n)$ 的通式會是什麼呢？以下為我們的研究。

先來描述「平面斜列防堵法」的排列規則：

平面斜列防堵法：

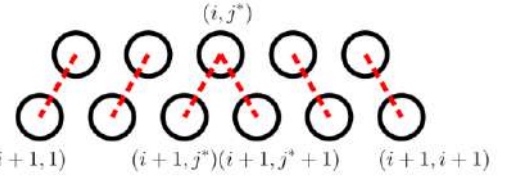
在 $\Delta_{n,\ell}$ 中，設第 i 列紅圓坐標為 $(i, j_s), 1 \leq s \leq m_i$ ，其中 $m_i = \min\{i, \ell\}$ ，則第 $i + 1$ 列的紅圓坐標為 $(i + 1, j'_s), 1 \leq s \leq m_{i+1}$ ，其中 $j_{m_i} + 1 \equiv j'_1 \pmod{i}, 1 \leq j'_1 \leq i$ 且 $j'_1 + s - 1 \equiv j'_s \pmod{i + 1}, 1 \leq j'_s \leq i + 1$ 。

定理 8：在 $\Delta_{n,\ell}$ 中 ($\ell \geq 2$)，令數列 $\langle a_k \rangle$ 滿足
$$\begin{cases} a_k = 1 + \left\lfloor \frac{\sum_{i=1}^{k-1} a_i + \frac{(\ell-1)(\ell-2)}{2}}{\ell-1} \right\rfloor, k \geq \ell + 1, \text{ 若} \\ a_1 = 1, a_2 = 2, \dots, a_\ell = \ell \end{cases}$$

$a_k \leq n < a_{k+1}$ ，則 $I_\ell(n) = k$ 。

【證明】

令 $S_i = \sum_{j=1}^i f(i, j)$ ，即第 i 列之最糟序列總和。



Claim1. $S_i = i^2$, $\forall 1 \leq i \leq \ell$ 且 $S_{i+1} \leq \ell + S_i + \left\lceil \frac{S_i}{i} \right\rceil$, $\forall i \geq \ell$

Proof. 顯然 $S_i = i^2$, $\forall 1 \leq i \leq \ell$ 是成立的，我們只需證明不等式中 $i \geq \ell$ 的情形。

令第 i 列之最糟序列中最小值為 $g(i, j^*)$ ，顯然 $g(i, j^*) \leq \left\lceil \frac{S_i}{i} \right\rceil$ 。

$$\begin{aligned} \text{則 } S_{i+1} &= \sum_{j=1}^{i+1} g(i+1, j) = \sum_{j=1}^{j^*} g(i+1, j) + \sum_{j=j^*+1}^{i+1} g(i+1, j) \\ &\leq \ell + \sum_{j=1}^{j^*} g(i, j) + \sum_{j=j^*}^i g(i, j) = \ell + S_i + g(i, j^*) \leq \ell + S_i + \left\lceil \frac{S_i}{i} \right\rceil \end{aligned}$$

Claim2. 若 $n = a_k + r$, $0 \leq r \leq a_{k+1} - a_k - 1$, $k \geq \ell$ ，則

當 $a_1 + a_2 + \dots + a_{k-1} + \frac{(\ell-1)(\ell-2)}{2} \equiv t \pmod{\ell-1}$, $t = 1, 2, \dots, \ell-1$, $k \geq 2$ 時，

$$S_n \leq nk + \ell \cdot r + (\ell-1-t)$$

Proof. 當 $n = \ell = a_\ell + 0$ ，即 $k = \ell, r = 0$

$$a_1 + a_2 + \dots + a_{\ell-1} + \frac{(\ell-1)(\ell-2)}{2} = 1 + 2 + \dots + (\ell-1) + \frac{(\ell-1)(\ell-2)}{2} = (\ell-1)^2 \equiv \ell-1 \pmod{\ell-1}$$

$$S_\ell = \ell^2 \leq \ell \cdot \ell + \ell \cdot 0 + (\ell-1 - (\ell-1)) = \ell^2 \quad \text{原式成立}$$

設 $n = a_k + r$, $0 \leq r \leq a_{k+1} - a_k - 1$, $k \geq \ell$ 時，原式成立 即

當 $a_1 + a_2 + \dots + a_{k-1} + \frac{(\ell-1)(\ell-2)}{2} \equiv t \pmod{\ell-1}$ 時， $S_n \leq nk + \ell \cdot r + (\ell-1-t)$ ，

此時

$$\begin{aligned} a_{k+1} - a_k &= \left(1 + \left\lceil \frac{a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2}}{\ell-1} \right\rceil \right) - \left(1 + \left\lceil \frac{a_1 + a_2 + \dots + a_{k-1} + \frac{(\ell-1)(\ell-2)}{2}}{\ell-1} \right\rceil \right) \\ &= \left\lceil \frac{a_k + (t-1)}{\ell-1} \right\rceil \end{aligned}$$

$$\text{則 } \left\lceil \frac{S_n}{n} \right\rceil \leq \left\lceil \frac{nk + \ell \cdot r + (\ell-1-t)}{n} \right\rceil = \left\lceil k + \frac{\ell \cdot r + (\ell-1-t)}{n} \right\rceil = k$$

上式原因如下：

$$\because 0 \leq r \leq \left\lceil \frac{a_k + (t-1)}{\ell-1} \right\rceil - 1$$

$$\therefore \ell-1-t \leq \ell \cdot r + (\ell-1-t) = (\ell-1) \cdot r + (\ell-1-t) + r \leq (\ell-1) \cdot \left(\left\lceil \frac{a_k + (t-1)}{\ell-1} \right\rceil - 1 \right) + (\ell-1-t) + r$$

$$\leq (\ell-1) \cdot \frac{a_k + (t-1)}{\ell-1} - t + r = a_k - 1 + r = n - 1 \Rightarrow 0 \leq \frac{\ell \cdot r + (\ell-1-t)}{n} < 1$$

分成 2 種 case 考慮 S_{n+1} :

(a) 當 $0 \leq r \leq a_{k+1} - a_k - 2 = \left\lfloor \frac{a_k + (t-1)}{\ell-1} \right\rfloor - 2$ 時, $n+1 = a_k + (r+1)$,

$$\text{則 } S_{n+1} \leq \ell + S_n + \left\lfloor \frac{S_n}{n} \right\rfloor \leq \ell + nk + \ell \cdot r + (\ell-1-t) + k = (n+1)k + \ell(r+1) + (\ell-1-t)$$

(b) 當 $r = a_{k+1} - a_k - 1 = \left\lfloor \frac{a_k + (t-1)}{\ell-1} \right\rfloor - 1$ 時, $n+1 = a_{k+1} + 0$

$$\begin{aligned} \text{則 } S_{n+1} &\leq \ell + S_n + \left\lfloor \frac{S_n}{n} \right\rfloor \leq \ell + nk + \ell \cdot r + (\ell-1-t) + k \\ &= (n+1)k + \ell \cdot (r+1) + (\ell-1-t) = (n+1)k + \ell \left\lfloor \frac{a_k + (t-1)}{\ell-1} \right\rfloor + (\ell-1-t) \end{aligned}$$

令 $a_k \equiv s \pmod{\ell-1}$, $s = 1, 2, \dots, \ell-1$, 即 $a_k = (\ell-1) \cdot p + s$, $p \in \mathbb{N}$,

$$\text{則 } a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2} \equiv t + s \pmod{\ell-1}$$

① 若 $t + s \leq \ell-1$, 則 $a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2} \equiv t + s \pmod{\ell-1}$

$$\text{且 } \left\lfloor \frac{a_k + (t-1)}{\ell-1} \right\rfloor = p = \frac{a_k - s}{\ell-1} \quad (\because a_k + (t-1) = (\ell-1) \cdot p + s + (t-1) \leq (\ell-1) \cdot p + (\ell-2))$$

$$\begin{aligned} \text{故 } S_{n+1} &\leq (n+1)k + \ell \left\lfloor \frac{a_k + (t-1)}{\ell-1} \right\rfloor + (\ell-1-t) = (n+1)k + \ell \cdot \frac{a_k - s}{\ell-1} + (\ell-1-t) \\ &= (n+1)k + (\ell-1) \cdot \frac{a_k - s}{\ell-1} + \frac{a_k - s}{\ell-1} + (\ell-1-t) = (n+1)k + (a_k - s) + (r+1) + (\ell-1-t) \\ &= (n+1)k + (a_k - s) + (n - a_k + 1) + (\ell-1-t) = (n+1)(k+1) + (\ell-1-t-s) \end{aligned}$$

② 若 $t + s > \ell-1$, 則 $a_1 + a_2 + \dots + a_k + \frac{(\ell-1)(\ell-2)}{2} \equiv t + s - (\ell-1) \pmod{\ell-1}$

$$\text{且 } \left\lfloor \frac{a_k + (t-1)}{\ell-1} \right\rfloor = p + 1 = \frac{a_k - s}{\ell-1} + 1 = \frac{a_k - s + \ell - 1}{\ell-1}$$

$$(\because a_k + (t-1) = (\ell-1) \cdot p + s + (t-1) > (\ell-1) \cdot p + (\ell-1) - 1 = (\ell-1)(p+1) - 1)$$

$$\begin{aligned} \text{故 } S_{n+1} &\leq (n+1)k + \ell \left\lfloor \frac{a_k + (t-1)}{\ell-1} \right\rfloor + (\ell-1-t) = (n+1)k + \ell \cdot \frac{a_k - s + \ell - 1}{\ell-1} + (\ell-1-t) \\ &= (n+1)k + (\ell-1) \cdot \frac{a_k - s + \ell - 1}{\ell-1} + \left(\frac{a_k - s + \ell - 1}{\ell-1} \right) + (\ell-1-t) \\ &= (n+1)k + (a_k - s + \ell - 1) + (n - a_k + 1) + (\ell-1-t) \\ &= (n+1)(k+1) + [\ell-1-(t+s-\ell+1)] \end{aligned}$$

(a)(b)皆成立 由數學歸納法得證

由 Claim1、Claim2 知, 若 $1 \leq n < \ell$, $g(n, j^*) \leq \left\lfloor \frac{S_n}{n} \right\rfloor = \left\lfloor \frac{n^2}{n} \right\rfloor = n = a_n$,

若 $n = a_k + r$, $0 \leq r \leq a_{k+1} - a_k - 1$, $k \geq \ell$,

$$g(n, j^*) \leq \left\lfloor \frac{S_n}{n} \right\rfloor \leq \left\lfloor \frac{nk + \ell \cdot r + (\ell - 1 - t)}{n} \right\rfloor = \left\lfloor k + \frac{\ell \cdot r + (\ell - 1 - t)}{n} \right\rfloor = k。而我們確實可以$$

「斜列防堵法」建構出一個 $\Delta_{n,\ell}$ 滿足 $g(n, j^*) = k$ ，因此若 $a_k \leq n < a_{k+1}$ ，則 $I_\ell(n) = k$ 。■

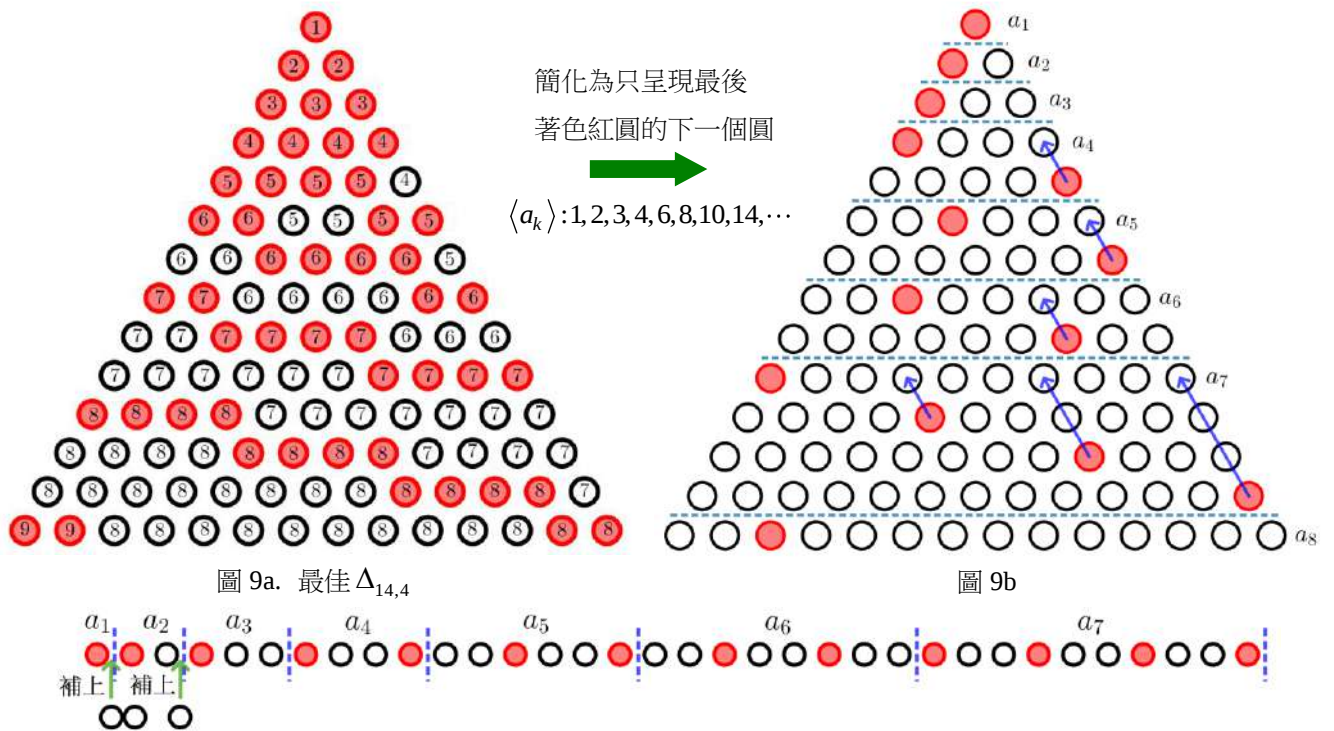
問題：定理 8 中的數列 $\langle a_k \rangle$ 之遞迴關係式如何得知？

分析：以圖 9a 做說明，它是一個以「斜列防堵法」建構出的最佳 $\Delta_{14,4}$ ，其第 $a_k \sim (a_{k+1} - 1)$ 列的最糟序列之最小值均為 k 。將圖 9b 中第 $a_1, a_2, a_3, a_4, a_5, a_6, a_7$ 列取出排成一列於圖 9c，並如圖 9b 所示補上紅圓，同時第 a_1, a_2 列須再分別補上 2 個與 1 個空圓，因此

此整列圓中共有 $\left\lceil \frac{a_1 + \dots + a_7 + (1+2)}{3} \right\rceil$ 個紅圓，又一行恰一個紅圓，即第 a_8 列前共有

$\left\lceil \frac{a_1 + \dots + a_7 + (1+2)}{3} \right\rceil$ 列，因此 $a_8 = 1 + \left\lceil \frac{a_1 + \dots + a_7 + (1+2)}{3} \right\rceil$ 。同理在最佳 $\Delta_{n,\ell}$ 中，取出的

第 $a_1 \sim a_{\ell-2}$ 列的圓必須要補上 $1+2+\dots+(\ell-2)$ 個，因此 $a_k = 1 + \left\lceil \frac{\sum_{i=1}^{k-1} a_i + \frac{(\ell-1)(\ell-2)}{2}}{\ell-1} \right\rceil$ 。



研究三： $\Delta_{n,\ell}^*$ 之最佳路徑最小化問題

在研究一與研究二中，我們已經將平面日式三角形的一般化情形全部解決，接著，便是要挑戰推廣至空間三角垛的情形，為了以下研究需要，我們再補充幾個名詞與符號定義。

名詞與符號定義 (研究三)

(一) n 階 ℓ 型日式三角垛(記作 $\Delta_{n,\ell}^*$)：將 $1+(1+2)+(1+2+3)+\cdots+(1+2+\cdots+n)$ 個球排成正三角

垛的形狀，使得對所有 $i=1,2,\cdots,n$ ，由上往下數的第 i 層有 $(1+2+\cdots+i)$

個球，且第 i 層塗成紅色的球數為 $\min\left\{\frac{i(i+1)}{2}, \ell\right\}$ 。

(二) 忍者路徑：一條路徑通過日式三角垛中最上層到最下層的球，其中每個球連到其下一層與之相鄰的三球之一。

(三) 最佳路徑：通過最多紅球數的忍者路徑。

(四) 最糟日式三角垛：由最上層到最下層，最佳路徑所通過的紅球數為最小的日式三角垛。

(五) $J_\ell^*(n)$ ： n 階 ℓ 型最糟日式三角垛中，由最上層到最下層，其最佳路徑所通過的紅球數。

(六) (i, j, k) ：日式三角垛中，由上往下數的第 i 層的三角形中，由頂點往底邊數的第 j 列，再由左往右數的第 k 個紅球之坐標。

(七) $f(i, j, k)$ ：由 $(1,1,1)$ 至 (i, j, k) 的最佳路徑所通過的紅球數。

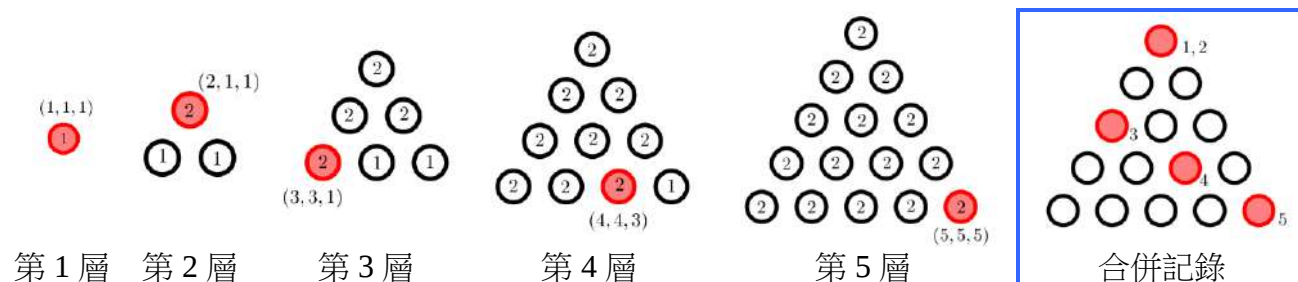
(八) 第 i 層之最佳陣列：
$$\begin{pmatrix} f(i,1,1) \\ f(i,2,1) & f(i,2,2) \\ f(i,3,1) & f(i,3,2) & f(i,3,3) \\ \vdots \\ f(i,i,1) & f(i,i,2) & \cdots & \cdots & f(i,i,i) \end{pmatrix}。$$

(九) 最佳陣列之次狀態陣列：給定一最佳陣列
$$\begin{pmatrix} f(i,1,1) \\ f(i,2,1) & f(i,2,2) \\ f(i,3,1) & f(i,3,2) & f(i,3,3) \\ \vdots \\ f(i,i,1) & f(i,i,2) & \cdots & \cdots & f(i,i,i) \end{pmatrix}，則其$$

次狀態陣列為
$$\begin{pmatrix} f(i,1,1) \\ \text{Max}\{f(i,1,1), f(i,2,1)\} & \text{Max}\{f(i,1,1), f(i,2,2)\} \\ \text{Max}\{f(i,2,1), f(i,3,1)\} & \text{Max}\{f(i,2,1), f(i,2,2), f(i,3,2)\} & \text{Max}\{f(i,2,2), f(i,3,3)\} \\ \vdots \\ f(i,i,1) & \text{Max}\{f(i,i,1), f(i,i,2)\} & \cdots & \cdots & \text{Max}\{f(i,i,i-1), f(i,i,i)\} & f(i,i,i) \end{pmatrix}$$

，記作
$$\begin{pmatrix} f(i,1,1) \\ f(i,2,1) & f(i,2,2) \\ f(i,3,1) & f(i,3,2) & f(i,3,3) \\ \vdots \\ f(i,i,1) & f(i,i,2) & \cdots & \cdots & f(i,i,i) \end{pmatrix}'。$$
 例如：
$$\begin{pmatrix} 2 \\ 2 & 2 \\ 2 & 1 & 1 \end{pmatrix}' = \begin{pmatrix} 2 \\ 2 & 2 \\ 2 & 2 & 2 \\ 2 & 2 & 1 & 1 \end{pmatrix}。$$

(十) $\Delta_{n,\ell}^*$ 記錄方式：(圓中數字代表 $f(i, j, k)$)



我們將 $\Delta_{n,\ell}^*$ 中 $f(i, j, k)$ 的求值規則整理於下：

$\Delta_{n,\ell}^*$ 中 $f(i, j, k)$ 的求值規則：

$$f(i, j, k) = \begin{cases} f(i-1, j, k) + c, & j = k = 1 \\ f(i-1, j-1, k) + c, & j = i, k = 1 \\ f(i-1, j-1, k-1) + c, & j = k = i \\ \text{Max}\{f(i-1, j-1, k), f(i-1, j, k)\} + c, & 2 \leq j \leq i-1, k = 1 \\ \text{Max}\{f(i-1, j-1, k-1), f(i-1, j, k)\} + c, & 2 \leq j = k \leq i-1 \\ \text{Max}\{f(i-1, j-1, k-1), f(i-1, j-1, k)\} + c, & j = i, 2 \leq k \leq i-1 \\ \text{Max}\{f(i-1, j-1, k-1), f(i-1, j-1, k), f(i-1, j, k)\} + c, & 2 \leq k < j \leq i-1 \end{cases}$$

其中當 (i, j, k) 為紅球時 $c=1$ ，反之 $c=0$ 。

由以上求值規則可得到以下性質：

性質 9：在 $\Delta_{n,\ell}^*$ 中，第 i 層最佳陣列總和＝第 $(i-1)$ 層最佳陣列之次狀態陣列總和 $+\ell$ 。

(一) $\ell=1$

先針對較小的 n 值做嘗試，找出 $J_1^*(n)$ 值，整理於下表：

n	1	2	3	4	5	6	7	8	9	10	11	12	...	...	125	126
$J_1^*(n)$	1	2	2	2	2	3	3	3	3	3	3	3	...	...	3	4

n	127	...	...	$2^{127}-3$	$2^{127}-2$	...	...	$2^{2^{127}-1}-3$	$2^{2^{127}-1}-2$	...	...	
$J_1^*(n)$	4	...	...	4	5	...	...	5	6	...	...	

類似於平面 $\Delta_{n,\ell}$ ，我們可以「空間斜列錯開法」建構出最糟日式三角垛(如圖 10)，其排列原則為每一層的紅球位置盡量錯開上一層紅球下方相鄰的三球，即若 (i, j, k) 為紅球，則 $(i+1, j', k')$ 亦為紅球，其中 $k+2 \equiv k' \pmod{j+1}$ ， $1 \leq k' \leq j+1$ 、 $j+1 + \left\lceil \frac{k+1}{j+1} \right\rceil \equiv j' \pmod{i+1}$ ， $1 \leq j' \leq i+1$ 。

此外， $J_1^*(n)$ 值增加的速度極為緩慢，和計算機科學中常見的迭代對數[4]特性很像，因此我們猜測它的公式的表示會類似於迭代對數。然而，在處理空間中三角垛的問題時，想要用代數的方法去處理證明是極繁瑣的，因此本作品我們選擇以較為直觀的方法證明我們的研究結果，也就是說，我們要說明「空間斜列錯開法」確實可建構出最糟日式三角垛(不會再更糟了)，接著再針對這樣的排列法找出 $J_1^*(n)$ 的通式。

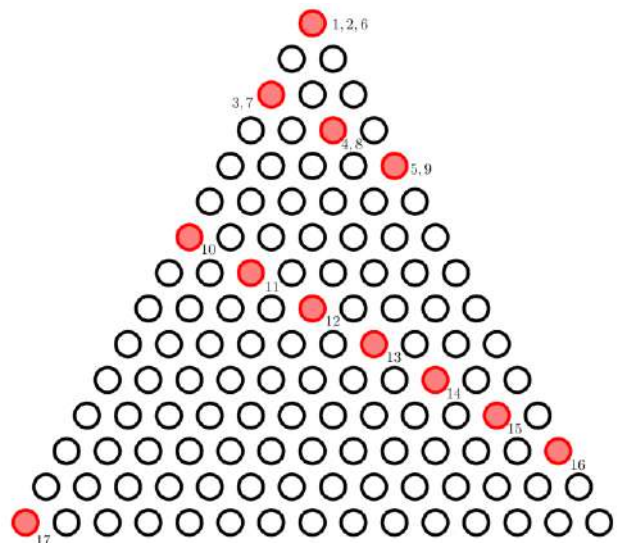


圖 10

下表 11 為以「空間斜列錯開法」排列的 $\Delta^*_{n,1}$ 每層紅球座標以及該層最佳陣列之最大值 k 。

k	1	2	3	4
紅 球 座 標	a_1 (1,1,1)	a_2 (2,1,1)	a_3 (6,1,1)	a_4 (126,1,1)
		(3,3,1)	(30,28,27)	
		(4,4,3)	(31,29,29)	
		(5,5,5)	(32,31,1)	
			(33,32,3)	
			(34,33,5)	
			(35,34,7)	
			(36,35,9)	
			(37,36,11)	
			(38,37,13)	
			(39,38,15)	
			(40,39,17)	
			(41,40,19)	
			(42,41,21)	
			(43,42,23)	
			(44,43,25)	
			(45,44,27)	
			(46,45,29)	
			(47,46,31)	
			(48,47,33)	
			(49,48,35)	
			(50,49,37)	
			(51,50,39)	
			(52,51,41)	
			(53,52,43)	
			(54,53,45)	
			(55,54,47)	
			(56,55,49)	
			(57,56,51)	
			(58,57,53)	
			(59,58,55)	
			(60,59,57)	
			(61,60,59)	
			(62,61,61)	
			(63,63,1)	
			(64,64,3)	
			(65,65,5)	
			(66,66,7)	
			(67,67,9)	
			(68,68,11)	
			(69,69,13)	
			(70,70,15)	
			(71,71,17)	
			(72,72,19)	
			(73,73,21)	
			(74,74,23)	
			(75,75,25)	
			(76,76,27)	
			(77,77,29)	
			(78,78,31)	
			(79,79,33)	
			(80,80,35)	
			(81,81,37)	
			(82,82,39)	
			(83,83,41)	
			(84,84,43)	
			(85,85,45)	
			(86,86,47)	
			(87,87,49)	
			(88,88,51)	
			(89,89,53)	
			(90,90,55)	
			(91,91,57)	
			(92,92,59)	
			(93,93,61)	
			(94,94,63)	
			(95,95,65)	
			(96,96,67)	
			(97,97,69)	
			(98,98,71)	
			(99,99,73)	
			(100,100,75)	
			(101,101,77)	
			(102,102,79)	
			(103,103,81)	
			(104,104,83)	
			(105,105,85)	
			(106,106,87)	
			(107,107,89)	
			(108,108,91)	
			(109,109,93)	
			(110,110,95)	
			(111,111,97)	
			(112,112,99)	
			(113,113,101)	
			(114,114,103)	
			(115,115,105)	
			(116,116,107)	
			(117,117,109)	
			(118,118,111)	
			(119,119,113)	
			(120,120,115)	
			(121,121,117)	
			(122,122,119)	
			(123,123,121)	
			(124,124,123)	
			(125,125,125)	

表 11

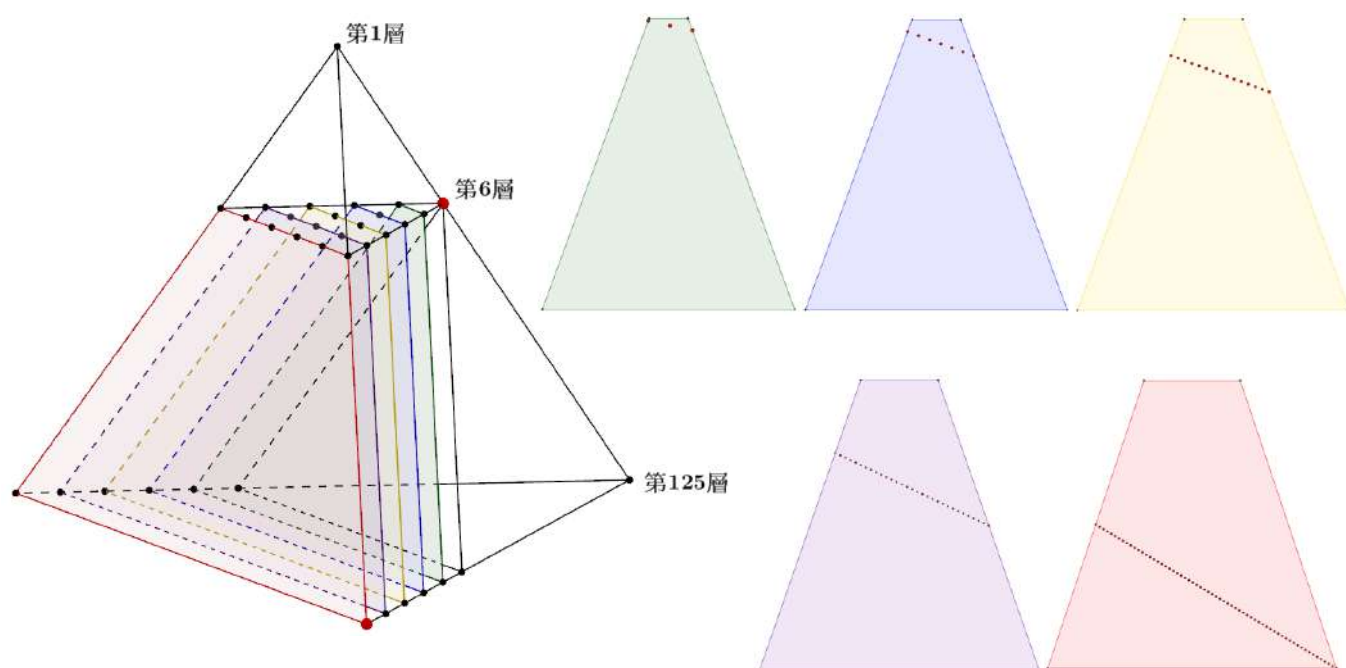


圖 12

圖 12 為以「空間斜列錯開法」排列的 $\Delta^*_{n,1}$ 之立體示意圖。其中第 6 層至第 125 層的紅球彼此之間均不存在任何忍者路徑。第 5 層最佳陣列所有數均為 2，即 $J_1^*(5) = 2$ ，當 $6 \leq n \leq 125$ 時， $J_1^*(n) = 3$ ，而第 125 層最佳陣列所有數均為 3，也就是說， $J_1^*(126) = 4$ 。

接下來我們要證明這個 $\Delta^*_{n,1}$ 是最糟日式三角垛，並且推導出 $J_1^*(n)$ 的通式。

定理 10：在 $\Delta^*_{n,1}$ 中，定義 $\log^{[3]}(n)$ 滿足 $\log^{[3]}(n) = \begin{cases} 0, & n < 1 \\ 1 + \log^{[3]}(\log_2(n+2)-1), & n \geq 1 \end{cases}$ ，則

$$J_1^*(n) = \log^{[3]}(n)。$$

【證明】

令 $S_i = \sum_{1 \leq k \leq j \leq i} f(i, j, k)$ ，即第 i 層之最佳陣列總和。

Claim1. 若將第 i 層之最佳陣列中的數由大至小排序得到 $a_1 \geq a_2 \geq \dots \geq a_{\frac{i(i+1)}{2}}$ ，則

$$S_{i+1} \geq 1 + S_i + a_1 + (a_1 + a_2 + a_4 + \dots + a_{\frac{i^2-i+2}{2}}), \quad \forall i \in N。$$

Proof. 註：圖 13a~13f 紅圓代表第 i 層之最佳陣列，藍點代表其次狀態陣列。

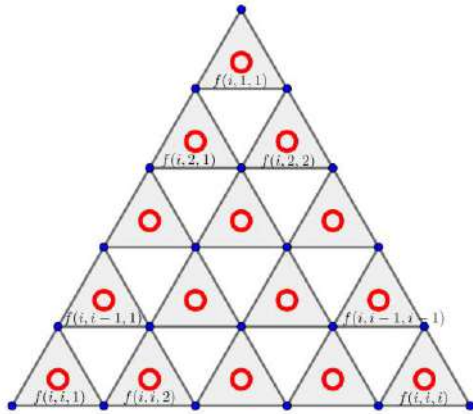


圖 13a

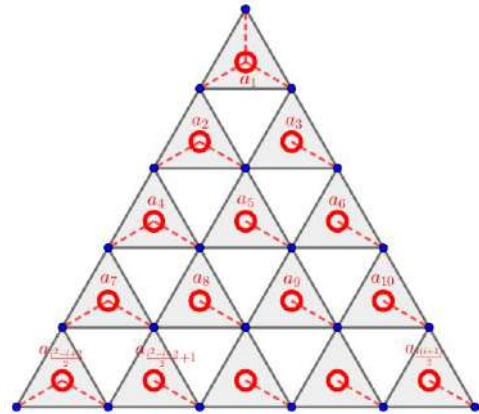


圖 13b

圖 13a 為第 i 層原始最佳陣列，圖 13b 為第 i 層原始最佳陣列重新排序後的新陣列，則

$$\left(\begin{array}{cccccc} f(i,1,1) & & & & & \\ & f(i,2,1) & f(i,2,2) & & & \\ & f(i,3,1) & f(i,3,2) & f(i,3,3) & & \\ & & \vdots & & & \\ f(i,i,1) & f(i,i,2) & \dots & \dots & \dots & f(i,i,i) \end{array} \right)' \text{總和} \geq \left(\begin{array}{cccccc} a_1 & & & & & \\ & a_2 & a_3 & & & \\ & a_4 & a_5 & a_6 & & \\ & & \vdots & & & \\ a_{\frac{i^2-i+2}{2}} & a_{\frac{i^2-i+2}{2}+1} & \dots & \dots & \dots & a_{\frac{i(i+1)}{2}} \end{array} \right)' \text{總和}。$$

原因如下：

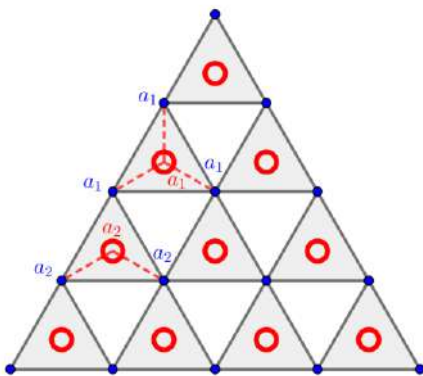


圖 13c

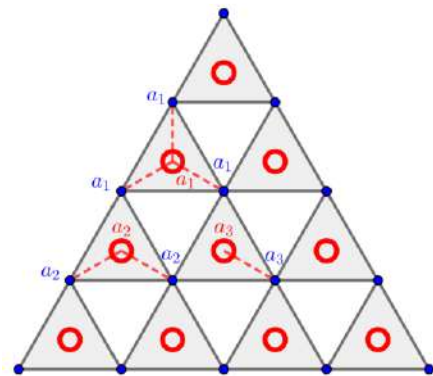


圖 13d

在 S_i 已知的狀況下，我們要考慮的是， $a_1 \sim a_{\frac{i(i+1)}{2}}$ 要如何排列其次狀態陣列總和才會最小。

將 $a_1 \sim a_{\frac{i(i+1)}{2}}$ 由大至小依序擺入第 i 層紅圓位置，在擺放的過程中可得兩個性質：

(1) **最佳置放**：若每一個數都盡量緊靠著排序在前方的數置放，所能控制的藍點數會最小。

(如圖 13c 中 $a_1 \sim a_2$ 共控制 5 個藍點，圖 13d 中 $a_1 \sim a_3$ 共控制 6 個藍點，依此類推。值得注意的是，**最佳置放的方法非唯一**，比如以圖 13e 或圖 13f 次序擺入皆為最佳置放。)

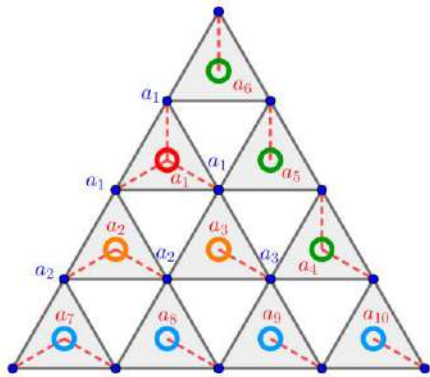


圖 13e

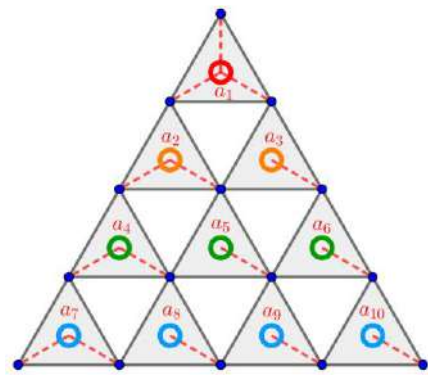


圖 13f

(2) **一路領先**：當 $a_1 \sim a_{\frac{i(i+1)}{2}}$ 以最佳置放模式依序擺放時，對於任意自然數 t ， $a_1 \sim a_t$ 所控

制的藍點數值總和必小於等於任意置放時所控制的藍點數值總和。

Proof. 觀察 $a_1 \sim a_t$ 以最佳置放與任意置放時所控制的藍點數值形成的數列(由大至小)：

最佳置放： $\langle b_n \rangle: a_1, a_1, a_1, a_2, a_2, a_3, a_4, a_4, a_5, a_6, \dots, a_t$

任意置放： $\langle b'_n \rangle: a_1, a_1, a_1, a_2, \dots, a_t$

令 $\forall t \in N$ ， $a_1 \sim a_t$ 以最佳置放與任意置放時分別控制 m_t 、 m'_t 個藍點

由性質(1)知， $m_t \leq m'_t, \forall t \in N$ ，可推得 $b_i \leq b'_i, \forall i \in N \Rightarrow \sum_{i=1}^{m_t} b_i \leq \sum_{i=1}^{m_t} b'_i \leq \sum_{i=1}^{m'_t} b'_i, \forall t \in N$ 。

由以上可知，第 i 層最佳陣列的次狀態陣列總和必大於等於其重新以最佳置放排序後新陣列的次狀態陣列總和，可得 $S_{i+1} = \sum_{1 \leq k \leq j \leq i+1} f(i+1, j, k) \geq 1 + S_i + a_1 + (a_1 + a_2 + a_4 + \dots + a_{\frac{i^2-i+2}{2}})$ 。

(重新排序後的新陣列的次狀態陣列總和為 $(a_1 + a_2 + \dots + a_{\frac{i(i+1)}{2}}) + 2a_1 + a_2 + a_4 + \dots + a_{\frac{i^2-i+2}{2}}$)

Claim2. 以「空間斜列錯開法」排列的 $\Delta^*_{n,1}$ 為最糟日式三角垛。

Proof. 由 Claim1 的證明過程知，當 S_i 已知且第 i 層最佳陣列為相同分割 $\left\{ a_1, a_2, \dots, a_{\frac{i(i+1)}{2}} \right\}$

，其中 $a_1 \geq a_2 \geq \dots \geq a_{\frac{i(i+1)}{2}}$ ，若最佳陣列為依照大至小由第 1 列至第 i 列的次序排列時，其次狀態陣列的總和為最小，此時 $S_{i+1} = 1 + S_i + a_1 + (a_1 + a_2 + a_4 + \dots + a_{\frac{i^2-i+2}{2}})$ ，又 S_i 為

定值，因此，當 $a_1, a_2, a_4, \dots, a_{\frac{i^2-i+2}{2}}$ 只由 $\left\lfloor \frac{S_i}{\frac{i(i+1)}{2}} \right\rfloor$ 與 $\left(\left\lfloor \frac{S_i}{\frac{i(i+1)}{2}} \right\rfloor - 1 \right)$ 組成時， S_{i+1} 能達到最小值。而第 $i+1$ 層之最佳陣列為只由 $\left\lfloor \frac{S_{i+1}}{\frac{(i+1)(i+2)}{2}} \right\rfloor$ 與 $\left(\left\lfloor \frac{S_{i+1}}{\frac{(i+1)(i+2)}{2}} \right\rfloor - 1 \right)$ 組成的分布時，由第 1 層至第 $i+1$ 層的最佳路徑所通過的紅球數為 $\left\lfloor \frac{S_{i+1}}{\frac{(i+1)(i+2)}{2}} \right\rfloor$ ，且不可能再少了！顯然，以「空間斜列錯開法」排列的 $\Delta^*_{n,1}$ 完全符合以上條件，即為最糟日式三角垛。

Claim3. 令數列 $\langle a_k \rangle$ 滿足 $\begin{cases} a_k = 2^{a_{k-1}+1} - 2, k \geq 2 \\ a_1 = 1 \end{cases}$ ，若 $a_k \leq n < a_{k+1}$ ，則 $J_1^*(n) = k$ 。

Proof. 由 Caim2 可知，以「空間斜列錯開法」排列的 $\Delta^*_{n,1}$ 為最糟日式三角垛，因此我們可以針對這個排列法去求 $J_1^*(n)$ 值。

由表 11 知 $J_1^*(i) = \begin{cases} J_1^*(i-1)+1, \text{第} i \text{層紅球座標為}(i,1,1) \\ J_1^*(i-1), \text{第} i \text{層紅球座標非}(i,1,1) \end{cases}$ ，令滿足 $J_1^*(n) = k$ 的最小 n 值

為 a_k ，則所有滿足 $J_1^*(n) = k-1$ 的 n 值為 $n = a_{k-1} \sim a_k - 1$ 。由第 a_{k-1} 層紅球座標 $(a_{k-1}, 1, 1)$ 開始，第 a_{k-1} 層至第 $a_k - 1$ 層的紅球排列會呈現數個斜列(如圖 12)。令第 t 個斜列的起始紅球座標為 $(i_t, j_t, 1)$ ，則該斜列座標變化為 $(i_t, j_t, 1) \rightarrow (i_t + j_t - 1, 2j_t - 1, 2j_t - 1)$ 且共有 j_t 層，且第 $t+1$ 個斜列的起始座標 $(i_t + j_t, 2j_t + 1, 1)$ ，當某斜列的最終座標層座標與列座標相等時，其下一層的紅球座標便是第 a_k 層的 $(a_k, 1, 1)$ 。我們可列出遞迴關係式

$$\begin{cases} i_{t+1} = i_t + j_t \\ j_{t+1} = 2j_t + 1 \end{cases}, i_1 = a_{k-1}, j_1 = 1, \text{可求得通式} \begin{cases} i_t = a_{k-1} + 2^t - t - 1 \\ j_t = 2^t - 1 \end{cases}, \text{當 } i_t + j_t - 1 = 2j_t - 1 \Rightarrow$$

$i_t = j_t \Rightarrow a_{k-1} + 2^t - t - 1 = 2^t - 1 \Rightarrow t = a_{k-1}$ ，即第 a_{k-1} 層至第 $a_k - 1$ 層共有 a_{k-1} 個斜列，每

個斜列 j_t 層，故 $a_k = a_{k-1} + j_1 + j_2 + \dots + j_{a_{k-1}} = a_{k-1} + \sum_{t=1}^{a_{k-1}} (2^t - 1) = \frac{2(2^{a_{k-1}} - 1)}{2 - 1} = 2^{a_{k-1}+1} - 2$ 。

由 Caim3 可知 $a_k = 2^{a_{k-1}+1} - 2 \Rightarrow a_{k-1} = \log_2(a_k + 2) - 1$ ，定義 $\log^{[3]}(n)$ 滿足遞迴關係

$$\log^{[3]}(n) = \begin{cases} 0, & n < 1 \\ 1 + \log^{[3]}(\log_2(n + 2) - 1), & n \geq 1 \end{cases}, \text{可得 } J_1^*(n) = \log^{[3]}(n)。$$

【 $\log^{[3]}(n)$ 的涵義】 n 可以執行幾次「加 2 後取以 2 為底的對數值再減 1」的運算後小於 1。

圖 14 為函數 $y = \log_2(x+2) - 1$ 的圖形，以 $n=10$ 為例， $\log_2(10+2)-1=a$ ， $\log_2(a+2)-1=b$ 、 $\log_2(b+2)-1=c < 1$ ，共執行 3 次運算(3 次迭代)，因此 $\log^{[3]}(10)=3$ ，即 $J_1^*(10)=3$ 。

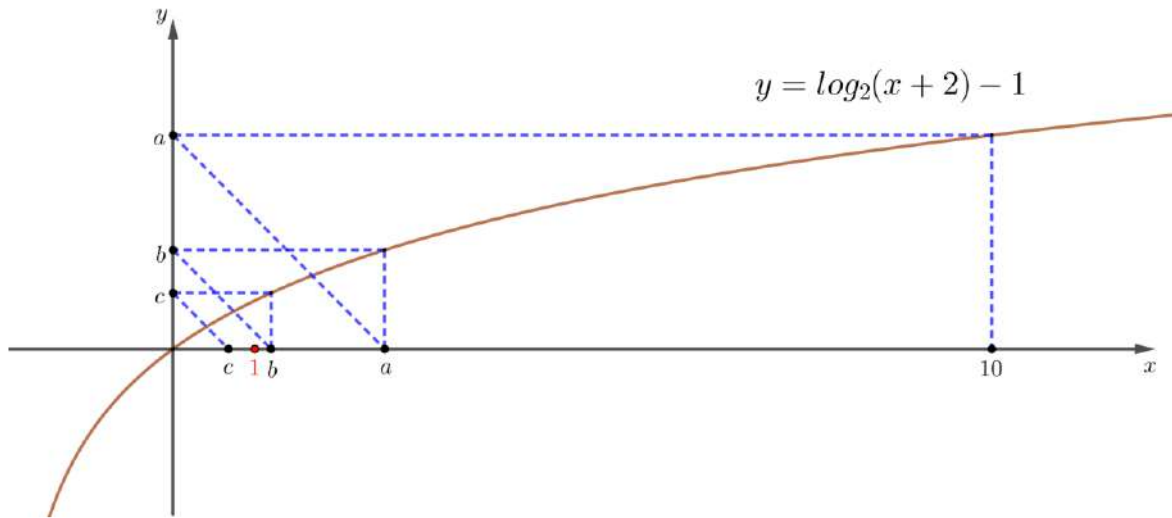


圖 14. $\log^{[3]}(n)$ 的迭代過程

(二) $\ell=2$

首先，先針對較小的 n 值做嘗試，找出 $J_2^*(n)$ 值，整理於下表：

n	1	2	3	4	5	6	7	8	9	10	11	12	...	...	66	67
$J_2^*(n)$	1	2	3	3	3	3	3	4	4	4	4	4	...	...	4	5

同理 $\Delta_{n,1}^*$ 的情形，我們可以「空間斜列錯開法」建構最糟 $\Delta_{n,2}^*$ ，並求得 $J_2^*(n)$ 的通式。

k	1	2	3	4				5
紅 球 座 標	a_1 (1,1,1)	a_2 (2,1)	a_3 (3,1,1)	a_4 (8,2,1)	(23,21,8)	(38,37,25)	(53,53,26)	a_5 (67,1,1)
			(4,3,2)	(9,4,1)	(24,22,11)	(39,38,28)	(54,54,29)	(68,3,2)
			(5,5,1)	(10,5,4)	(25,23,14)	(40,39,31)	(55,55,32)	(69,5,1)
			(6,6,4)	(11,7,1)	(26,24,17)	(41,40,34)	(56,56,35)	(70,6,4)
			(7,7,7)	(12,8,4)	(27,25,20)	(42,41,37)	(57,57,38)	(71,7,7)
				(13,9,7)	(28,26,23)	(43,42,40)	(58,58,41)	(72,9,2)
				(14,10,10)	(29,27,26)	(44,43,43)	(59,59,44)	(73,10,5)
				(15,12,2)	(30,29,1)	(45,45,2)	(60,60,47)	(74,11,8)
				(16,13,5)	(31,30,4)	(46,46,5)	(61,61,50)	(75,12,11)
				(17,14,8)	(32,31,7)	(47,47,8)	(62,62,53)	(76,14,1)
				(18,15,11)	(33,32,10)	(48,48,11)	(63,63,56)	⋮
				(19,16,14)	(34,33,13)	(49,49,14)	(64,64,59)	⋮
				(20,17,17)	(35,34,16)	(50,50,17)	(65,65,62)	
				(21,19,2)	(36,35,19)	(51,51,20)	(66,66,65)	
				(22,20,5)	(37,36,22)	(52,52,23)		

表 15. 「空間斜列錯開法」排列的 $\Delta_{n,2}^*$ 每層最後著色紅球座標以及該層最佳陣列之最大值 k

我們將公式推導結果寫成定理 11：(詳細證明於研究日誌)

$$\text{定理 11: 在 } \Delta_{n,2}^* \text{ 中, 數列 } \langle a_k \rangle \text{ 滿足 } \begin{cases} a_k = a_{k-1} + \left\lfloor \frac{\sum_{i=1}^{a_{k-1}-b_{k-1}+1} r_i^{(b_{k-1})}}{2} \right\rfloor, k \geq 2, \text{ 若 } a_k \leq n < a_{k+1}, \text{ 則 } J_2^*(n) = k. \\ a_1 = 1, b_1 = 1 \end{cases}$$

$$\text{其中 } b_k = \begin{cases} 1, \text{ 當 } \sum_{i=1}^{a_{k-1}-b_{k-1}+1} r_i^{(b_{k-1})} \text{ 為偶數} \\ 2, \text{ 當 } \sum_{i=1}^{a_{k-1}-b_{k-1}+1} r_i^{(b_{k-1})} \text{ 為奇數} \end{cases}, \text{ 數列 } \langle r_t^{(s)} \rangle \text{ 滿足 } \begin{cases} r_t^{(s)} = t + (s-1) + \left\lfloor \frac{\sum_{i=1}^{t-1} r_i^{(s)}}{2} \right\rfloor, t \geq 2, s = 1, 2. \\ r_1^{(s)} = s \end{cases}$$

由表 15 知，第 a_k 層紅球座標可能為 $(a_k, 1, 1)$ 或 $(a_k, 2, 1)$ ，因此其公式需透過 2 個相異 b_k 值與 2 個相異數列 $\langle r_t^{(s)} \rangle$ 來表示，且當 ℓ 越來越大時，其紅球座標更難由簡易的遞迴關係式來表示，而是須透過多重的迭代過程才能決定，因此我們評估，將 $J_\ell^*(n)$ 表示成通式的數學價值已不足夠，若是可以設計出一套迭代演算法，配合電腦程式設計，即可有效率的求出 $J_\ell^*(n)$ 。

(三) $\ell \in N$

當 $\ell=1, 2$ 時，我們證明了「空間斜列錯開法」確實可排列出最糟日式三角垛，這個結論在 $\ell \in N$ 當然也會成立。以下我們先定義「空間斜列錯開法」的紅球坐標表示，其排列原則是每一層的紅球位置盡量錯開上一層紅球下方相鄰的三球。

空間斜列錯開法

在 $\Delta_{n,\ell}^*$ 中，設第 i 層紅球座標依序為 $(i, j_s, k_s), 1 \leq s \leq m_i$ ，其中 $m_i = \min \left\{ \frac{i(i+1)}{2}, \ell \right\}$ 。

令 $(i', j_s', k_s') = (i+1, j_{m_i}+1, k_{m_i}+s+1), 1 \leq s \leq m_{i+1}$ ，進行以下迭代步驟，

(1) 當 $i' < j_s'$ 時， $(i', j_s', k_s') \Rightarrow (i', j_s' - i', k_s')$ 再執行步驟(2)。否則直接步驟(2)。

(2) 當 $j_s' < k_s'$ 時， $(i', j_s', k_s') \Rightarrow (i', j_s' + 1, k_s' - j_s')$ 回步驟(1)。

重複進行(1)(2)至 $i' \geq j_s' \geq k_s'$ 為止。此時 $(i', j_s', k_s'), 1 \leq s \leq m_{i+1}$ 為第 $i+1$ 層的紅球坐標。

定理 12：以「空間斜列錯開法」排列的 $\Delta_{n,\ell}^*$ 為最糟日式三角垛。

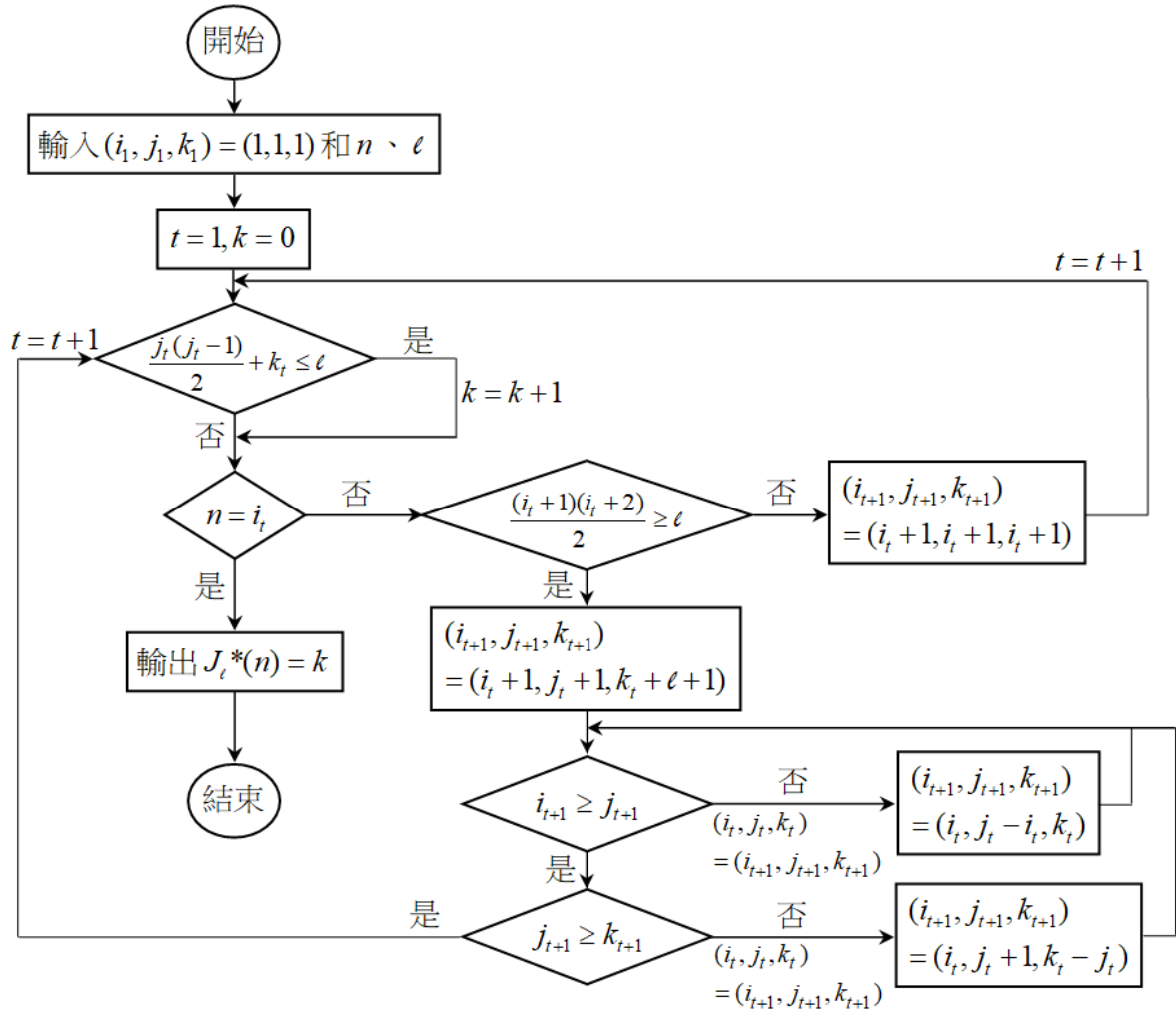
【證明】同理定理 10 中 Caim1、Caim2 之證明可得證。(詳細證明於研究日誌)

【求 $J_\ell^*(n)$ 之演算法】

給定 n, ℓ ，由第 1 層紅球座標 $(1, 1, 1)$ 開始，利用上述「空間斜列錯開法」之迭代過程，可以決定第 i 層的 m_i 個紅球坐標分別為 $(i, j_s, k_s), 1 \leq s \leq m_i$ ，考慮 $s = m_i$ ，即第 i 層的第 m_i 個著色的

球坐標 (i, j_{m_i}, k_{m_i}) ，則 $J_\ell^*(i) = \begin{cases} J_\ell^*(i-1) + 1, \text{ 當 } \frac{j_{m_i}(j_{m_i}-1)}{2} + k_{m_i} \leq \ell \\ J_\ell^*(i-1), \text{ 當 } \frac{j_{m_i}(j_{m_i}-1)}{2} + k_{m_i} > \ell \end{cases}, i \geq 2, \text{ 其中 } J_\ell^*(1) = 1. \text{ 進行以}$

上迭代關係式後即可求得 $J_\ell^*(n)$ 。(程式碼於附錄)



以下為電腦程式跑出來的部分結果：

ℓ	1	2	3	4	5	6	7	8
a_1	1	1	1	1	1	1	1	1
a_2	2	2	2	2	2	2	2	2
a_3	6	3	3	3	3	3	3	3
a_4	126	8	5	4	4	4	4	4
a_5	$\vdots$	67	12	7	6	5	5	5
a_6	$\vdots$	1991116568762	132	20	12	9	8	7
a_7	$\vdots$	$\vdots$	$\vdots$	429	46	21	14	11
a_8	$\vdots$	$\vdots$	$\vdots$	$\vdots$	24718	174	45	25
a_9	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	3169	161
a_{10}	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	1490886909

表 16. 若 $a_k \leq n < a_{k+1}$ ，則 $J_\ell^*(n) = k$ 。

研究四： $\Delta_{n,\ell}^*$ 之最糟路徑最大化問題

研究四中，我們將問題改成：「找出 k 的最小值，保證在每一個日式三角形垛中，有一條包含至多 k 個紅色球的忍者路徑。」為了研究需要，補充幾個名詞與符號定義於下：

名詞與符號定義 (研究四)

(一)最糟路徑：通過最少紅球數的忍者路徑。

(二)最佳日式三角垛：由最上層到最下層，最糟路徑所通過的紅球數為最大的日式三角垛。

(三) $I_\ell^*(n)$ ： n 階 ℓ 型最佳日式三角垛中，由最上層到最下層，其最糟路徑所通過的紅球數。

(四) $g(i, j, k)$ ：由 $(1,1,1)$ 至 (i, j, k) 的最糟路徑所通過的紅球數。

(五)第 i 層之最糟陣列：
$$\begin{pmatrix} g(i,1,1) \\ g(i,2,1) & g(i,2,2) \\ g(i,3,1) & g(i,3,2) & g(i,3,3) \\ \vdots \\ g(i,i,1) & g(i,i,2) & \dots & \dots & g(i,i,i) \end{pmatrix}。$$

(六)最糟陣列之次狀態陣列：給定一最糟陣列 $\begin{pmatrix} g(i,1,1) \\ g(i,2,1) & g(i,2,2) \\ g(i,3,1) & g(i,3,2) & g(i,3,3) \\ \vdots \\ g(i,i,1) & g(i,i,2) & \dots & \dots & g(i,i,i) \end{pmatrix}$ ，則其

次狀態陣列為
$$\begin{pmatrix} g(i,1,1) \\ \min\{g(i,1,1), g(i,2,1)\} & \min\{g(i,1,1), g(i,2,2)\} \\ \min\{g(i,2,1), g(i,3,1)\} & \min\{g(i,2,1), g(i,2,2), g(i,3,2)\} & \min\{g(i,2,2), g(i,3,3)\} \\ \vdots \\ g(i,i,1) & \min\{g(i,i,1), g(i,i,2)\} & \dots & \dots & \min\{g(i,i,i-1), g(i,i,i)\} & g(i,i,i) \end{pmatrix}，$$

記作 $\begin{pmatrix} g(i,1,1) \\ g(i,2,1) & g(i,2,2) \\ g(i,3,1) & g(i,3,2) & g(i,3,3) \\ \vdots \\ g(i,i,1) & g(i,i,2) & \dots & \dots & g(i,i,i) \end{pmatrix}'$ 。例如：
$$\begin{pmatrix} 2 \\ 2 & 2 \\ 2 & 1 & 1 \\ 2 & 1 & 1 & 1 \end{pmatrix}' = \begin{pmatrix} 2 \\ 2 & 2 \\ 2 & 1 & 1 \\ 2 & 1 & 1 & 1 \end{pmatrix}。$$

我們將 $\Delta_{n,\ell}^*$ 中 $g(i, j, k)$ 的求值規則整理於下：

$\Delta_{n,\ell}^*$ 中 $g(i, j, k)$ 的求值規則：

$$g(i, j, k) = \begin{cases} g(i-1, j, k) + c, & j = k = 1 \\ g(i-1, j-1, k) + c, & j = i, k = 1 \\ g(i-1, j-1, k-1) + c, & j = k = i \\ \min\{g(i-1, j-1, k), g(i-1, j, k)\} + c, & 2 \leq j \leq i-1, k = 1 \\ \min\{g(i-1, j-1, k-1), g(i-1, j, k)\} + c, & 2 \leq j = k \leq i-1 \\ \min\{g(i-1, j-1, k-1), g(i-1, j-1, k)\} + c, & j = i, 2 \leq k \leq i-1 \\ \min\{g(i-1, j-1, k-1), g(i-1, j-1, k), g(i-1, j, k)\} + c, & 2 \leq k < j \leq i-1 \end{cases}$$

其中當 (i, j, k) 為紅球時 $c=1$ ，反之 $c=0$ 。

由以上求值規則可得到以下性質：

性質 13：在 $\Delta_{n,\ell}^*$ 中，第 i 層最糟陣列總和 = 第 $(i-1)$ 層最糟陣列之次狀態陣列總和 + ℓ 。

當 $\ell=1$ 或 $\ell=2$ 時， $I_1^*(n)=I_2^*(n)=1$ 是顯然成立的，因為每顆球的下層均連接 3 顆球，當每層的紅球數小於 3 時，除了第一層起始紅球外，必存在一條忍者路徑完全不再經過其他紅球即可到達最下層。我們將此結論寫成定理 14 和定理 15。

定理 14：在 $\Delta_{n,1}^*$ 中， $I_1^*(n)=1$ 。

定理 15：在 $\Delta_{n,2}^*$ 中， $I_2^*(n)=1$ 。

接著，我們考慮 $\ell=3$ 的情形。先針對較小的 n 值做嘗試，找出 $I_3^*(n)$ 值，整理於下表：

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$I_3^*(n)$	1	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

由上表知，除了 $I_3^*(1)=1$ 外， $I_3^*(n)=2, \forall n \geq 2$ 。建構最佳日式三角垛的方式和「平面斜列防堵法」相似，其排列原則是盡可能在空球下方安排紅球，避免路徑一路經過太多空球，其中每層紅球位置安排，必須從三角形底邊往頂點排列，這個結論可由定理 16 中的 Claim1 得知。

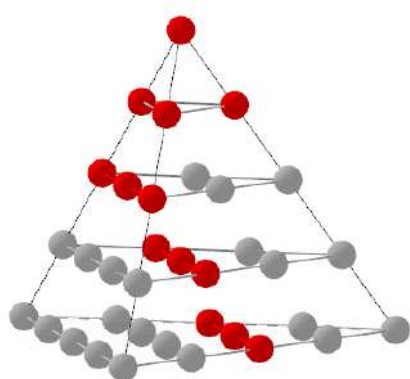
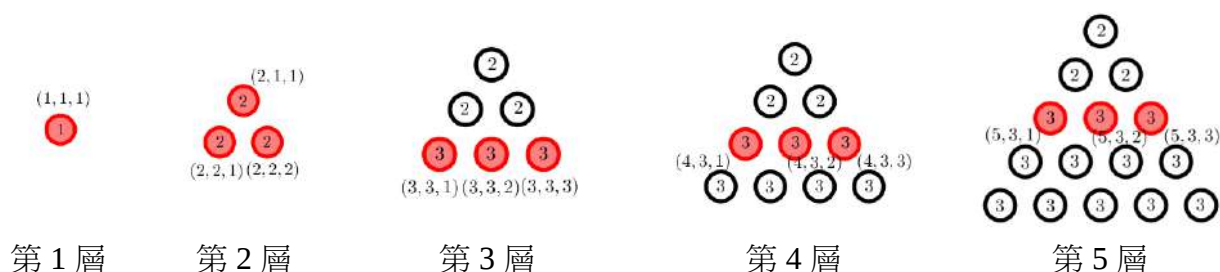


圖 17

圖 17 是以「空間斜列防堵法」排列而成的最佳日式三角垛。可觀察到，縱使我們很想要中斷任何一條從第三層開始就沒有通過紅球的路徑，但這是辦不到的，因為從第三層起，每層留下來的空球數太多，僅以 3 顆紅球並不足以阻擋第三層起全空球路徑的發生。因此， $I_3^*(n)=2, \forall n \geq 2$ 。當然，我們需要嚴謹的證明。

定理 16：在 $\Delta_{n,3}^*$ 中， $I_3^*(n)=\begin{cases} 1, n=1 \\ 2, n \geq 2 \end{cases}$ 。

【證明】

當 $n=1$ 時， $I_3^*(n)=1$ 顯然成立。考慮 $n \geq 2$ 的情形，顯然 $I_3^*(n) \geq 2$ 。

令 $S_i = \sum_{1 \leq k \leq j \leq i} g(i, j, k)$ ，即第 i 層之最糟陣列總和。

Claim1. 若將第 i 層之最糟陣列中的數由小至大排序得到 $a_1 \leq a_2 \leq \dots \leq a_{\frac{i(i+1)}{2}}$ ，則

$$S_{i+1} \leq 3 + S_i + a_1 + (a_1 + a_2 + a_4 + \dots + a_{\frac{i^2-i+2}{2}}), \quad \forall i \in \mathbb{N}。$$

Proof. 將定理 10 中的 Claim1. 證明過程反向思考，即同理可證。(詳細證明於研究日誌)

Claim2. $S_n \leq \frac{3n(n+1)}{2} - 3, \forall n \geq 2$

Proof. 當 $n=2$ 時， $S_2 = 6 \leq \frac{3 \cdot 2 \cdot 3}{2} - 3$ 原式成立

$$\text{設 } S_n \leq \frac{3n(n+1)}{2} - 3 = 2 \times 3 + 3 \times \left(\frac{n(n+1)}{2} - 3 \right) \Rightarrow 2 \leq \frac{S_n}{\frac{n(n+1)}{2}} < 3$$

$$\text{則 } S_{n+1} \leq 3 + S_n + 2 + \underbrace{(2 + 2 + 3 + \dots + 3)}_{n \text{ 項}}$$

$$\leq 3 + \frac{3n(n+1)}{2} - 3 + 6 + 3 \times (n-2) = \frac{3(n+1)(n+2)}{2} - 3$$

由數學歸納法得證

$$\text{由 Claim2 知，} \left\lfloor \frac{S_n}{\frac{n(n+1)}{2}} \right\rfloor \leq \left\lfloor \frac{\frac{3n(n+1)}{2} - 3}{\frac{n(n+1)}{2}} \right\rfloor = \left\lfloor 3 - \frac{6}{n(n+1)} \right\rfloor = 2 \Rightarrow I_3^*(n) = 2, n \geq 2。 \blacksquare$$

再考慮 $\ell=4$ 的情形。我們可以「空間斜列防堵法」建構最佳 $\Delta_{n,4}^*$ 。表 18 記錄其每層最先著色紅球座標以及該層最糟陣列之最小值 k 。

k	1	2	3
紅	a_1 (1,1,1)	a_2 (2,2,2)	a_3 (5,2,2)
球		(3,3,3)	(6,6,5)
座		(4,3,2)	(7,7,2)
標			(8,7,5)
			(9,8,2)
			(10,8,6)
			(11,9,3)
			(12,9,8)
			(13,10,5)
			(14,11,2)
			$\vdots$

表 18

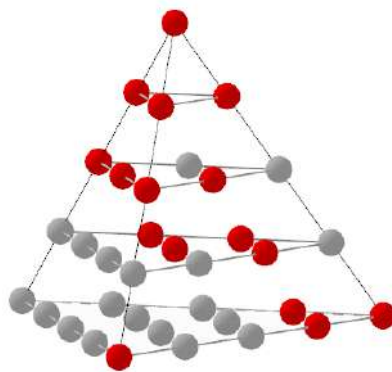


圖 19

由圖 19 知，從第三層起，我們盡可能將紅球安排於空球下方，在第 5 層時，成功的阻斷全空球路徑的發生，因此 $I_4^*(5) = 3$ 。但從第六層起，每層留下來的空球數太多，僅以 4 顆紅球並不足以阻擋第六層後全空球路徑的發生。因此 $I_4^*(n) = 3, \forall n \geq 5$ 。

由以上例子，我們發現一個重要的事實，當 n 足夠大時， $I_\ell^*(n)$ 必會呈現定值。

空間斜列防堵法

在 $\Delta_{n,\ell}^*$ 中，設第 i 層紅球坐標依序為 $(i, j_s, k_s), 1 \leq s \leq m_i$ ，其中 $m_i = \min \left\{ \frac{i(i+1)}{2}, \ell \right\}$ 。

令 $(x, y, z) = (i, j_{m_i}, k_{m_i} - 1)$ ，進行以下迭代步驟，

(1) 當 $y < 1$ 時， $(x, y, z) \Rightarrow (x, y+x, z+x)$ 再執行步驟(2)。否則直接步驟(2)。

(2) 當 $z < 1$ 時， $(x, y, z) \Rightarrow (x, y-1, z+y-1)$ 。回步驟(1)。

重複進行(1)(2)至 $x \geq y \geq z \geq 1$ 為止。

令 $(i', j'_s, k'_s) = (x+1, y+1, z+2-s), 1 \leq s \leq m_{i+1}$ ，進行以下迭代步驟，

(3) 當 $j'_s < 1$ 時， $(i', j'_s, k'_s) \Rightarrow (i', j'_s + i', k'_s + i')$ 。

(4) 當 $k'_s < 1$ 時， $(i', j'_s, k'_s) \Rightarrow (i', j'_s - 1, k'_s + j'_s - 1)$ 。回步驟(3)。

重複進行(3)(4)至 $i' \geq j'_s \geq k'_s \geq 1$ 為止。此時 $(i', j'_s, k'_s), 1 \leq s \leq m_{i+1}$ 為第 $i+1$ 層的紅球坐標。

定理 17：以「空間斜列防堵法」排列的 $\Delta_{n,\ell}^*$ 為最佳日式三角垛。

【證明】

令 $S_i = \sum_{1 \leq k \leq j \leq i} g(i, j, k)$ ，即第 i 層之最糟陣列總和。

Claim. 若將第 i 層之最糟陣列中的數由小至大排序得到 $a_1 \leq a_2 \leq \dots \leq a_{\frac{i(i+1)}{2}}$ ，則

$$S_{i+1} \leq \ell + S_i + a_1 + (a_1 + a_2 + a_4 + \dots + a_{\frac{i^2-i+2}{2}}), \quad \forall i \in N \text{ 且 } 1+2+\dots+(i+1) \geq \ell。$$

Proof. 同理定理 16 中的 Claim1 的證明過程可證。(詳細證明於研究日誌)

由 Claim 的證明過程知，當 S_i 已知，若最糟陣列為依照小至大由第 1 列至第 i 列的次序排列時，其次狀態陣列的總和為最大，此時 $S_{i+1} = \ell + S_i + a_1 + (a_1 + a_2 + a_4 + \dots + a_{\frac{i^2-i+2}{2}})$ ，又

當 $a_1, a_2, a_4, \dots, a_{\frac{i^2-i+2}{2}}$ 只由 $\left\lfloor \frac{S_i}{\frac{i(i+1)}{2}} \right\rfloor$ 與 $\left(\left\lfloor \frac{S_i}{\frac{i(i+1)}{2}} \right\rfloor + 1 \right)$ 組成時， S_{i+1} 能達到最大值。而第 $i+1$ 層

之最糟陣列為只由 $\left\lfloor \frac{S_{i+1}}{\frac{(i+1)(i+2)}{2}} \right\rfloor$ 與 $\left(\left\lfloor \frac{S_{i+1}}{\frac{(i+1)(i+2)}{2}} \right\rfloor + 1 \right)$ 組成的分布時，由第 1 層至第 $i+1$ 層的最糟

路徑所通過的紅球數為 $\left\lfloor \frac{S_{i+1}}{\frac{(i+1)(i+2)}{2}} \right\rfloor$ ，且不可能再多了！顯然，以「空間斜列防堵法」排列

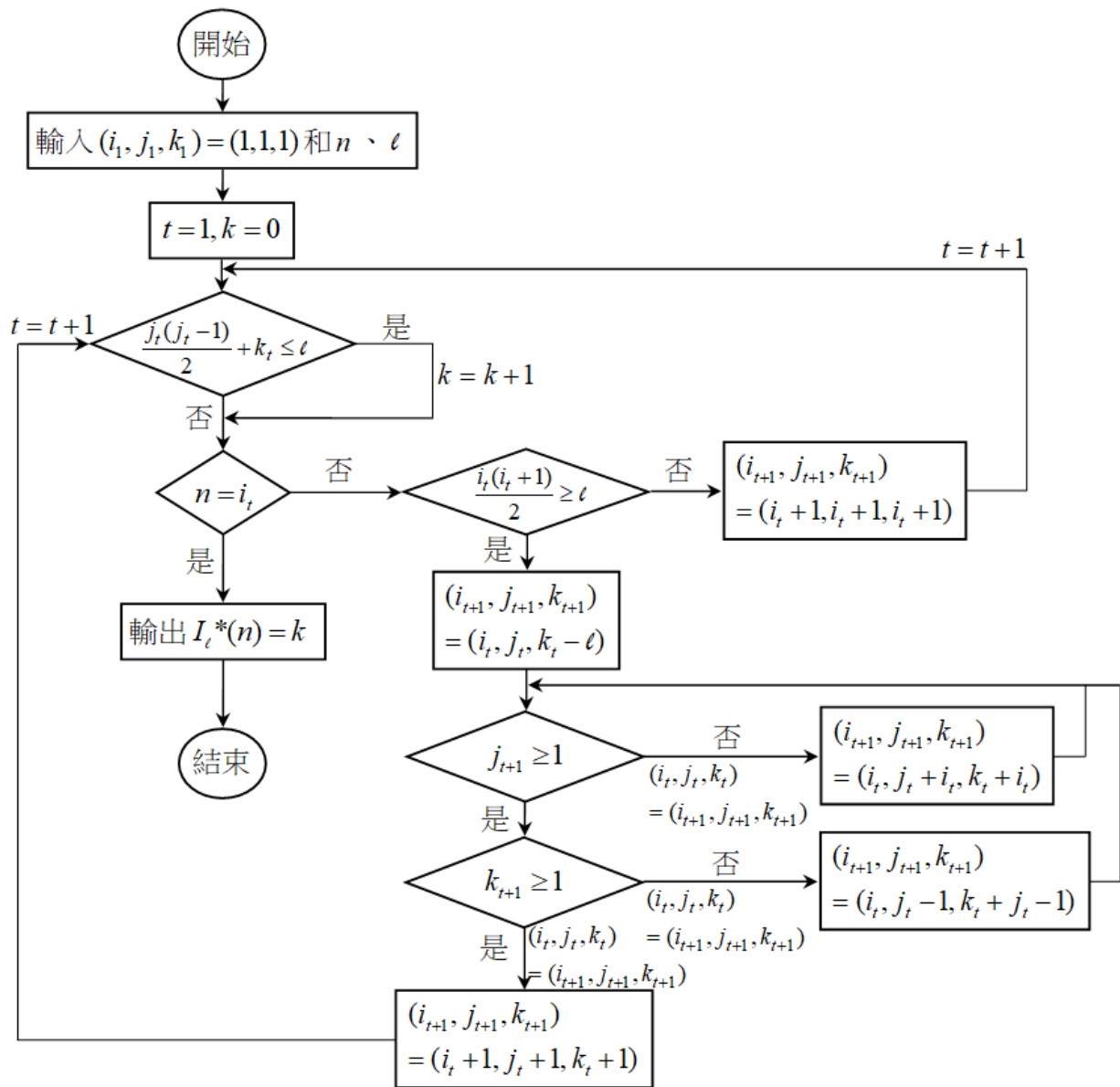
的 $\Delta_{n,\ell}^*$ 完全符合以上條件，即為最佳日式三角垛。■

【求 $I_\ell^*(n)$ 之演算法】

給定 n 、 ℓ ，由第 1 層紅球座標 $(1,1,1)$ 開始，利用上述「空間斜列防堵法」之迭代過程，可以決定第 i 層的 m_i 個紅球座標分別為 (i, j_s, k_s) , $1 \leq s \leq m_i$ ，考慮 $s=1$ ，即第 i 層的第 1 個著色的球

座標 (i, j_1, k_1) ，則 $I_\ell^*(i) = \begin{cases} I_\ell^*(i-1)+1, & \text{當 } \frac{j_1(j_1-1)}{2} + k_1 \leq \ell \\ I_\ell^*(i-1) & \text{當 } \frac{j_1(j_1-1)}{2} + k_1 > \ell \end{cases}, i \geq 2, I_\ell^*(1)=1$ 。進行以上迭代關係

式後即可求得 $I_\ell^*(n)$ 。(程式碼於附錄)



我們將電腦程式跑出來的部分結果整理如下：

ℓ	1	2	3	4	5	6	7	8	9	10	11
a_1	1	1	1	1	1	1	1	1	1	1	1
a_2	×	×	2	2	2	2	2	2	2	2	2
a_3	×	×	×	5	4	3	3	3	3	3	3
a_4	×	×	×	×	×	6	5	5	5	4	4
a_5	×	×	×	×	×	×	13	9	7	6	6
a_6	×	×	×	×	×	×	×	×	21	11	10
a_7	×	×	×	×	×	×	×	×	×	×	×

表 20. 若 $a_k \leq n < a_{k+1}$ ，則 $I_\ell^*(n) = k$ 。

【 $I_\ell^*(n)$ 定值性質之探討】

由表 20 知，當 n 足夠大時， $I_\ell^*(n)$ 會呈現定值，即存在 $i, c \in \mathbb{N}$ ，使得 $I_\ell^*(n) = c, \forall n \geq i$ 。那麼， i 的最小值會發生在何時呢？我們觀察 $\ell = 4$ 的情形。

圖 21 為紅球以「空間斜列防堵法」排列的 $\Delta_{6,4}^*$ ：

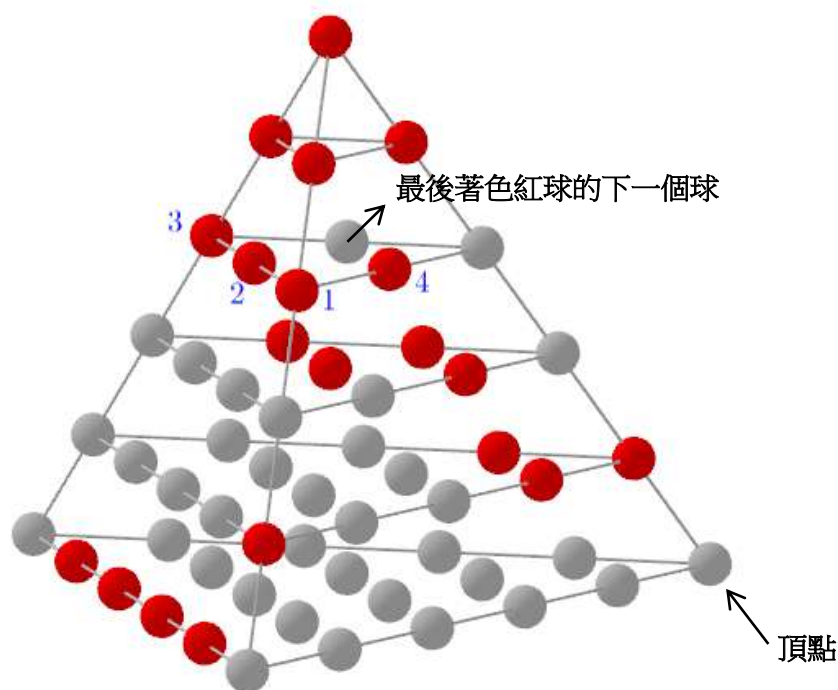


圖 21

【說明】

1. 紅球著色的次序為該層三角形底邊往頂點方向排列。
2. 由最後著色紅球的下一個球起至該層三角形頂點之球所成的集合代表 塗完紅球後剩餘的球，以 E_n 表示，而其個數記作 $n(E_n) = e_n$ 。

以圖 21 為例， $E_3 = \{(3, 2, 1), (3, 1, 1)\}$ 且 $e_3 = 2$ 。同理 $e_4 = 1$ 、 $e_5 = 14$ 。

3. 補球時機：著色至該層三角形頂點後回到底邊著色或剛好著色至頂點結束。如圖 21 中的第 2 層與第 5 層。

將數據整理於下表，並觀察其每層塗完紅球後所剩餘的球數間之關係：

層數 (n)	該層總球數 S_n	最後著色紅球的下一 個球坐標(i_n, j_n, k_n)	塗完紅球後剩 餘的球數 e_n	下一層與 E_n 相 鄰的球數 e_n'	$I_\ell^*(n)$ 值
1	1	(1,1,1)	1	3	1
2	3	(2,2,2)	3	6	2
3	6	(3,2,1)	2	5	2
4	10	(4,1,1)	1	3	2
5	15	(5,5,4)	14	20	3
6	21	(6,6,1)	16	23	3
7	28	(7,6,4)	19	26	3
8	36	(8,7,1)	22	30	3
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

表 22. 紅球以「空間斜列防堵法」排列的 $\Delta_{n,4}^*$

由表 22 可知， $I_4^*(n)=3, \forall n \geq 5$ ，也就是從 $n=5$ 起， $I_4^*(n)$ 開始呈現定值 3。同時我們可觀察到，讓 $I_\ell^*(n)$ 不再增加的關鍵是，每層塗完紅球後剩餘的球數是否在沒有補球的狀態下遞增(含相等)。經過研究，我們得到了以下結論：

定理 18：在 $\Delta_{n,\ell}^*$ 中，存在 $i, c \in \mathbb{N}$ ，使得 $I_\ell^*(n)=c, \forall n \geq i$ ，而 i 的最小值為最小正整數 i 滿足不等式 $j \geq \ell-1$ ，其中 (i, j, k) 為以「空間斜列防堵法」排列的 $\Delta_{n,\ell}^*$ 之第 i 層的最後著色紅球的下一個球坐標。

【證明】

當 $S_n \leq \ell$ 時，代表該層全塗成紅球，補球後得到 $e_n = S_n$ 。此時 $I_\ell^*(n) = I_\ell^*(n-1) + 1, n \geq 2$ 。

當 $S_n > \ell$ 時，令第 n 層最後著色紅球的下一個球坐標為 (i_n, j_n, k_n) ，則 $e_n = \frac{j_n(j_n-1)}{2} + k_n$ ，

且 $e_n' = e_n + j_n + 1$ 。

分成 2 種 case 考慮：

(1) 當 $e_n' > \ell$ 時，第 $n+1$ 層塗完紅球後剩餘 $e_n' - \ell$ 個球。

(2) 當 $e_n' \leq \ell$ 時，第 $n+1$ 層塗了 e_n' 個紅球後，必須再從該層底邊往頂點方向塗 $(\ell - e_n')$ 個紅球，因此第 $n+1$ 層塗完紅球後剩餘 $[S_{n+1} - (\ell - e_n')]$ 個球。

由(1)(2) 可整理成遞迴關係式 $e_{n+1} = \begin{cases} e_n' - \ell, & e_n' > \ell \\ e_n' - \ell + S_{n+1}, & e_n' \leq \ell \end{cases}$ ，其中 $e_n' = e_n + j_n + 1$ 。

由以上可知，當 $e_n' \leq \ell$ 時， $e_n' - \ell$ 需補上 S_{n+1} 個球後才會是 e_{n+1} ，此時第 $n+1$ 層的最糟陣列之最小值會增加 1，故 $I_\ell^*(n+1) = I_\ell^*(n) + 1, n \geq 1$ 。當 e_n 在沒有補球的狀態下開始遞增，代表 $I_\ell^*(n)$ 值不再增加，即開始呈現定值，此時， $e_{n+1} \geq e_n \Rightarrow e_n + j_n + 1 - \ell \geq e_n \Rightarrow j_n \geq \ell - 1$ 。■

定理 18 所得到的結論，雖然無法讓我們明確得知 i 的最小值，但它對我們接下來要找出 i 的最小值的上下界有極大的幫助。

定理 19：在 $\Delta^*_{n,\ell}$ 中，存在 $i, c \in \mathbb{N}$ ，使得 $I_\ell^*(n) = c, \forall n \geq i$ ，而 i 的最小值 $\min_\ell(i)$ 滿足

$$\ell - 1 \leq \min_\ell(i) \leq \ell - 2 + \sum_{k=1}^{\ell-2} \left\lceil \frac{\ell-2}{k} \right\rceil。$$

【證明】

Claim1. $\min_\ell(i) \geq \ell - 1$

Proof. 由定理 18 可知， $\min_\ell(i)$ 為最小正整數 i 滿足不等式 $j \geq \ell - 1$ ，其中 (i, j, k) 為以「空間斜列防堵法」排列的 $\Delta^*_{n,\ell}$ 之第 i 層的最後著色紅球的下一個球坐標。

顯然， $i \geq j \geq \ell - 1$ ，因此 $\min_\ell(i) \geq \ell - 1$ 。

Claim2. $\min_\ell(i) \leq \ell - 2 + \sum_{k=1}^{\ell-2} \left\lceil \frac{\ell-2}{k} \right\rceil$

Proof. 觀察以「空間斜列防堵法」排列的 $\Delta^*_{n,\ell}$ ，依序擴張 n 值，可發現每當紅球塗到第 n 層頂點時， $I_\ell^*(n)$ 便會加 1。也就是說，假設紅球由上層往下層排列，第 t 次塗到頂點的層數為 a_t ，則 $I_\ell^*(a_t) = t$ 。而 $I_\ell^*(n)$ 產生定值 c 的時機為最後一次紅球塗到頂點的層數，也就是第 a_c 層，因此 $\min_\ell(i) = a_c$ 。為了求出 $\min_\ell(i)$ 的上界，我們考慮倒數第二次紅球塗到頂點的層數 a_{c-1} ，並分別求出 a_{c-1} 的上界與 $a_c - a_{c-1}$ 的上界。

(1) $a_{c-1} \leq \ell - 2$

我們可以用反證法證明它。假設 $a_{c-1} > \ell - 2$ ，即 $a_{c-1} \geq \ell - 1$ ，觀察第 a_{c-1} 層紅球排列的所有可能情形。圖 23a 與圖 23b 所示是兩個極端的排法。

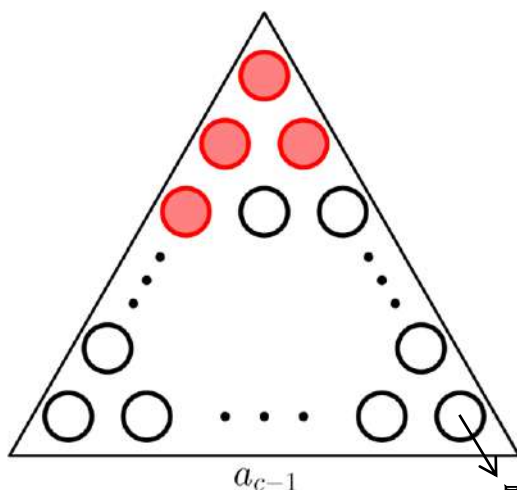


圖 23a

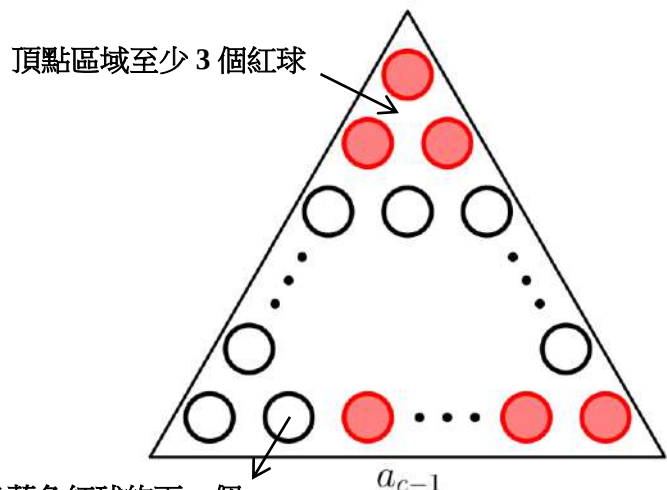


圖 23b

由上圖可知，無論第 a_{c-1} 層紅球如何排列，它的最後著色紅球的下一個球座標 (i, j, k) 都會滿足 $j = a_{c-1} \geq \ell - 1$ ，此時符合定理 18 產生定值的時機，與已知矛盾。

$$(2) a_c - a_{c-1} \leq \sum_{k=1}^{\ell-2} \left\lceil \frac{\ell-2}{k} \right\rceil$$

為了找出 $a_c - a_{c-1}$ 的上界，我們鎖定極端狀態，即 $a_{c-1} = \ell - 2$ 且最後著色紅球恰為該層頂點的情形。例如 $\ell = 9$ 的情形便是滿足此極端情形(表 24)。我們可得知，第 a_{c-1} 層到第 $a_c - 1$ 層的 e_n 值會逐步遞減直至 $e_n' \leq \ell$ 為止，且其遞減的量會遞增。

事實上，此遞減量決定於最後著色紅球的下一個球座標 (i_n, j_n, k_n) 中的 j_n 之值。

說明如下：

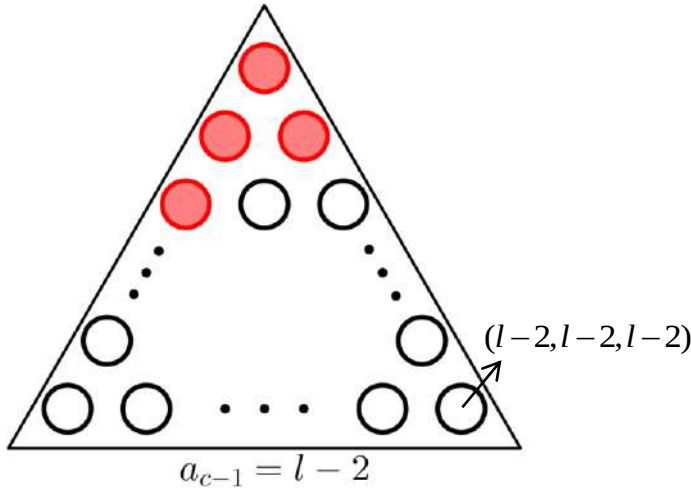


圖 23c

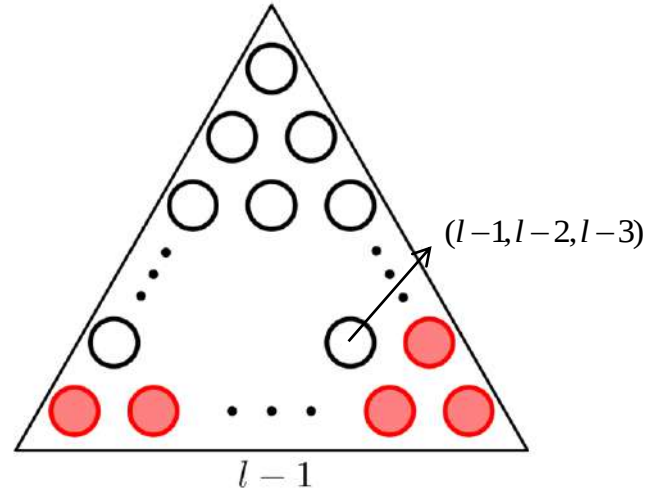


圖 23d

考慮 $\ell - 2 = a_{c-1} \leq n \leq a_c - 2$ ，假設第 n 層最後著色紅球的下一個球座標為 (i_n, j_n, k_n) ，則 $e_n' = e_n + j_n + 1$ ，且 $e_{n+1} = e_n + j_n + 1 - \ell = e_n - (\ell - j_n - 1)$ 。當 $n = \ell - 2$ 且最後著色紅球恰為該層頂點時(圖 23c)， $j_n = \ell - 2$ ，故 $e_{n+1} = e_n - 1$ (圖 23d)。

又因為 $e_{n+1} < e_n \Rightarrow j_{n+1} \leq j_n = \ell - 2 \Rightarrow e_{n+2} < e_{n+1}$ ，依此類推，可得 $j_{n+1} \leq j_n$ 且

$e_{n+1} < e_n, \forall a_{c-1} \leq n \leq a_c - 2$ 。因此， e_n 的遞減量 $\ell - j_n - 1$ 會遞增。而當 $j_n = \ell - 2$ 時，遞減量為 1，但最多只會有 $\ell - 2$ 個 n 值會滿足 $j_n = \ell - 2$ ，因此遞減量為 1 的情形最多只會有 $\ell - 2$ 次。當 $j_n = \ell - 3$ 時，遞減量為 2，但最多只會有 $\ell - 3$ 個 n 值會滿足

$j_n = \ell - 3$ ，因此遞減量為 2 的情形最多只會有 $\left\lceil \frac{\ell-3}{2} \right\rceil$ 次。依此類推，可得

$$a_c - a_{c-1} \leq \left\lceil \frac{\ell-2}{1} \right\rceil + \left\lceil \frac{\ell-3}{2} \right\rceil + \cdots + \left\lceil \frac{1}{\ell-2} \right\rceil = \left\lceil \frac{\ell-2}{1} \right\rceil + \left\lceil \frac{\ell-2}{2} \right\rceil + \cdots + \left\lceil \frac{\ell-2}{\ell-2} \right\rceil = \sum_{k=1}^{\ell-2} \left\lceil \frac{\ell-2}{k} \right\rceil$$

(此處使用到上下取整轉換公式，即 $\left\lceil \frac{a}{b} \right\rceil = \left\lfloor \frac{a+b-1}{b} \right\rfloor, \forall a, b \in \mathbb{Z}$)

由(1)(2)可知， $\min_{\ell}(i) = a_c = a_{c-1} + a_c - a_{c-1} \leq \ell - 2 + \sum_{k=1}^{\ell-2} \left\lceil \frac{\ell-2}{k} \right\rceil$ 。 ■

層數 (n)	該層總球數 S_n	最後著色紅球的下 一個球坐標(i_n, j_n, k_n)	塗完紅球後剩 餘的球數 e_n	下一層與 E_n 相 鄰的球數 e'_n	$I_\ell^*(n)$ 值
1	1	(1,1,1)	1	3	1
2	3	(2,2,2)	3	6	2
3	6	(3,3,3)	6	10	3
4	10	(4,1,1)	1	3	3
5	15	(5,4,3)	9	14	4
6	21	(6,3,2)	5	9	4
7	28	(7,7,7)	28	36	5
8	36	(8,7,6)	27	35	5
9	45	(9,7,5)	26	34	5
10	55	(10,7,4)	25	33	5
11	66	(11,7,3)	24	32	5
12	78	(12,7,2)	23	31	5
13	91	(13,7,1)	22	30	5
14	105	(14,6,6)	21	28	5
15	120	(15,6,4)	19	26	5
16	136	(16,6,2)	17	24	5
17	153	(17,5,5)	15	21	5
18	171	(18,5,2)	12	18	5
19	190	(19,4,3)	9	14	5
20	210	(20,3,2)	5	9	5
21	231	(21,21,21)	231	253	6
22	253	(22,22,13)	244	267	6
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

表 24. 紅球以「空間斜列防堵法」排列的 $\Delta_{n,9}^*$

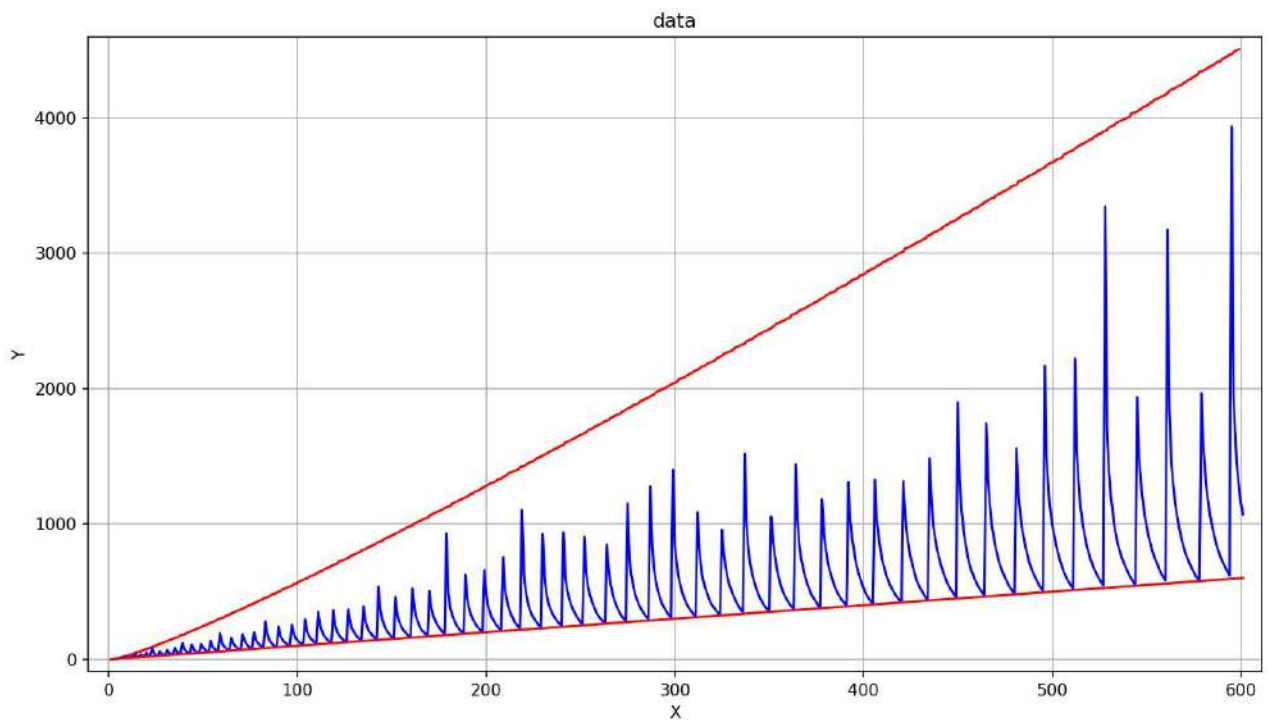


圖 25. $\ell = 1 \sim 600$ 時 $\min_\ell(i)$ 值之折線圖及其上下界

研究五： $\ell=1$ 高維空間情形之探討

在這個研究中，我們要探討日式三角垛的高維情形。為了方便起見，我們將各維空間的球都稱為球，並將 n 階 ℓ 型日式三角垛的 m 維情形記作 $\Delta_{n,\ell}^{[m]}$ ，我們的研究目的是，求出 $\Delta_{n,1}^{[m]}$ 中的 $J_1^{[m]}(n)$ 之值。以下為日式三角垛在高維空間(m 維空間)中的定義。

日式三角垛在高維空間(m 維空間)中的定義 (記作 $\Delta_{n,\ell}^{[m]}$)

n 階 ℓ 型日式三角垛在 m 維空間中的球之座標為 $(x_1, x_2, \dots, x_m)$ ，其中

$n \geq x_1 \geq x_2 \geq \dots \geq x_m \geq 1$ ，且 $x_1 = i$ 共有 C_{m-1}^{i+m-2} 個球，其中塗成紅色的球數為 $\min\{C_{m-1}^{i+m-2}, \ell\}$ 。

忍者路徑為一條路徑通過 $x_1 = 1$ 到 $x_1 = m$ 的球，其中 $x_1 = i$ 的每顆球連接到 $x_1 = i+1$ 中與之相鄰的 m 球之一。

最糟 $\Delta_{n,\ell}^{[m]}$

由 $x_1 = 1$ 到 $x_1 = m$ ，最佳路徑所通過的紅球數為最小的 $\Delta_{n,\ell}^{[m]}$ ，其紅圓數記作 $J_\ell^{[m]}(n)$ 。

(一) $m = 4$ 、 $\ell = 1$

類似於「3 維空間斜列錯開法」，定義「4 維空間斜列錯開法」如下：

4 維空間斜列錯開法

在 $\Delta_{n,1}^{[4]}$ 中，設 4 維空間座標為 (x_1, x_2, x_3, x_4) ， $x_1 = i_1$ 的紅球座標為 (i_1, i_2, i_3, i_4) 。

令 $(i_1', i_2', i_3', i_4') = (i_1 + 1, i_2 + 1, i_3 + 1, i_4 + 2)$ ，進行以下迭代步驟，

(1) 當 $i_1' < i_2'$ 時， $(i_1', i_2', i_3', i_4') \Rightarrow (i_1', i_2' - i_1', i_3', i_4')$ 再執行步驟(2)。否則直接步驟(2)。

(2) 當 $i_2' < i_3'$ 時， $(i_1', i_2', i_3', i_4') \Rightarrow (i_1', i_2' + 1, i_3' - i_2', i_4')$ 再執行步驟(3)。否則直接步驟(3)。

(3) 當 $i_3' < i_4'$ 時， $(i_1', i_2', i_3', i_4') \Rightarrow (i_1', i_2', i_3' + 1, i_4' - i_3')$ 回步驟(1)。

重複進行(1)(2)(3)至 $i_1' \geq i_2' \geq i_3' \geq i_4'$ 為止。此時 (i_1', i_2', i_3', i_4') 為 $x_1 = i_1 + 1$ 的紅球坐標。

舉例來說， $x_1 = 1$ 的紅球座標為 $(1, 1, 1, 1)$ ，令 $(i_1', i_2', i_3', i_4') = (1 + 1, 1 + 1, 1 + 1, 1 + 2) = (2, 2, 2, 3)$ ，依照上述迭代步驟，其迭代過程為 $(2, 2, 2, 3) \Rightarrow (2, 2, 3, 1) \Rightarrow (2, 3, 1, 1) \Rightarrow (2, 1, 1, 1)$ ，因此 $x_1 = 2$ 的紅球座標為 $(2, 1, 1, 1)$ 。重新令 $(i_1', i_2', i_3', i_4') = (2 + 1, 1 + 1, 1 + 1, 1 + 2) = (3, 2, 2, 3)$ ，再進行一次上述迭代步驟，其迭代過程為 $(3, 2, 2, 3) \Rightarrow (3, 2, 3, 1) \Rightarrow (3, 3, 1, 1)$ ，因此 $x_1 = 3$ 的紅球座標為 $(3, 3, 1, 1)$ 。依此類推，下表為以「4 維空間斜列錯開法」排列的 $\Delta_{n,1}^{[4]}$ 之紅球座標以及最佳陣列之最大值 k 。

k	1	a_2 2	b_1	3	4
紅 球 座 標	$(1,1,1,1)$ a_1	$(2,1,1,1)$	a_3 (14,1,1,1)	(83,72,63,1)	(8204,8193,8191,1)
		(3,3,1,1)	b_2 (15,3,1,1) c_1	$\vdots$	$\vdots$
		(4,4,3,1)	b_2 (16,4,3,1) c_2	(145,134,125,125)	(16394,16383,16381,16381)
		(5,5,4,3)	(17,5,4,3)	(146,135,127,1)	(16395,16384,16383,1)
		(6,6,5,5)	(18,6,5,5)	$\vdots$	$\vdots$
		(7,7,7,1)	(19,7,7,1) c_3	(272,261,253,253)	(32777,32766,32765,32765)
		(8,8,8,3)	(20,8,8,3)	(273,262,255,1)	(32778,32767,32767,1)
		(9,9,9,5)	(21,9,9,5)	$\vdots$	$\vdots$
		(10,10,10,7)	(22,10,10,7)	(527,516,509,509)	(65544,65533,65533,65533)
		(11,11,11,9)	(23,11,11,9)	(528,517,511,1)	(65545,65535,1,1)
		(12,12,12,11)	(24,12,12,11)	$\vdots$	$\vdots$
		(13,13,13,13)	(25,13,13,13)	(1038,1027,1021,1021)	$\vdots$
			b_3 (26,15,1,1)	$\vdots$	$\vdots$
			(27,16,3,1)	(1039,1028,1023,1)	$\vdots$
			$\vdots$	$\vdots$	$\vdots$
			(29,18,5,5)	(2061,2050,2045,2045)	$\vdots$
			(30,19,7,1)	(2062,2051,2047,1)	$\vdots$
			$\vdots$	$\vdots$	$\vdots$
			(36,25,13,13)	(4108,4097,4093,4093)	$\vdots$
			(37,26,15,1)	(4109,4098,4095,1)	$\vdots$
			$\vdots$	$\vdots$	$\vdots$
			(51,40,29,29)	(8203,8192,8189,8189)	$\vdots$
			(52,41,31,1)	$\vdots$	$\vdots$
			$\vdots$	$\vdots$	$\vdots$
			(82,71,61,61)	$\vdots$	$\vdots$

我們仔細觀察所有紅球座標規律，發現每個座標間都存在著某種意義與關係。整理如下：

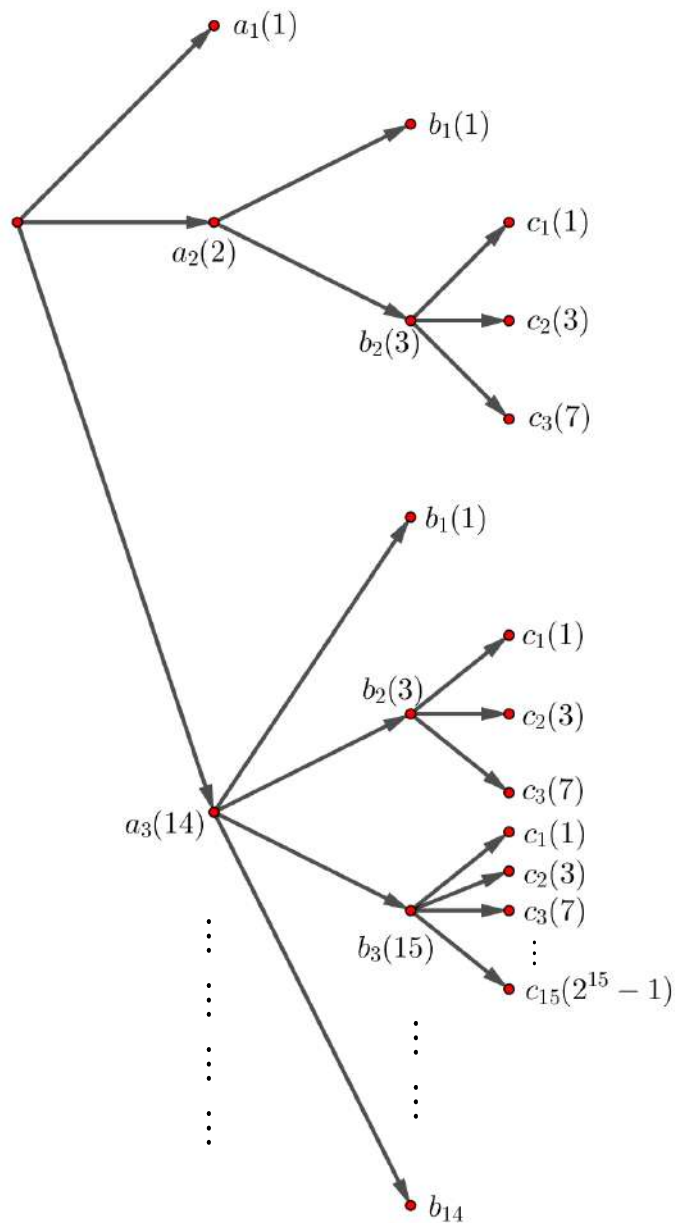
- (1) $\Delta_{n,1}^{[4]}$ 中的起始紅球座標為 $(a_1, 1, 1, 1)$ ，其中 $a_1 = 1$ 。當下一個出現 $(\bigcirc, 1, 1, 1)$ 的座標時， k 值增加 1，即數列 $\langle a_k \rangle$ 滿足 $(a_1, 1, 1, 1) \rightarrow (a_2, 1, 1, 1) \rightarrow (a_3, 1, 1, 1) \rightarrow \dots$ ，由上表知 $a_1 = 1$ 、 $a_2 = 2$ 、 $a_3 = 14$ 、 $\dots$ 。
- (2) 觀察不同的 k 值所有的紅球座標，以 $k = 3$ 為例，它的第一個紅球座標為 $(a_3, 1, 1, 1)$ ，其中 $a_3 = 14$ ，代表著 $k = 3$ 中，可區分成 14 個不同顏色區塊(如上表中的黃→綠→藍→ $\dots$)，而每一個顏色區塊的起始紅球座標皆型如 $(\bigcirc, \square, 1, 1)$ ，將這 14 個型如 $(\bigcirc, \square, 1, 1)$ 的座標中 $\square$ 依序標示成 $b_1, b_2, \dots, b_{14}$ ，即數列 $\langle b_k \rangle$ 滿足 $(a_i, b_1, 1, 1) \rightarrow (\bigcirc', b_2, 1, 1) \rightarrow (\bigcirc'', b_3, 1, 1) \rightarrow \dots$ 共 a_i 項，由上表可知 $b_1 = 1$ 、 $b_2 = 3$ 、 $b_3 = 15$ 、 $b_4 = 65535$ 、 $\dots$ ，事實上 $b_k = {}^k 2 - 1$ 。

【註】 ${}^b a$ 稱為迭代冪次(四級運算)，定義為 ${}^b a = a^{a^{a^{\cdot^{\cdot^{\cdot^a}}}}}$ ，即 a 連續取冪於自己 b 次，亦可以符號

$a[4]b$ 表示。

- (3) 觀察同色區塊中所有的紅球座標，以 $k = 3$ 中第 2 個綠色區塊為例，它的第一個紅球座標為 $(15, b_2, 1, 1)$ ，其中 $b_2 = 3$ ，代表著綠色區塊可再細分成 3 個小區域，每個小區域的起始紅球座標皆型如 $(\bigcirc, \square, \Delta, 1)$ ，將這 3 個型如 $(\bigcirc, \square, \Delta, 1)$ 的座標中 Δ 依序標示成 c_1, c_2, c_3 ，由上表可知 $c_1 = 1$ 、 $c_2 = 3$ 、 $c_3 = 7$ ，代表每個小區域的紅球數量，即綠色區域共有 $c_1 + c_2 + c_3 = 11$ 個紅球。依此類推可得數列 $\langle c_k \rangle$ 滿足 $(\bigcirc, b_i, c_1, 1) \rightarrow (\bigcirc', \square, c_2, 1) \rightarrow (\bigcirc'', \square, c_3, 1) \rightarrow \dots$ 共 b_i 項，事實上 $c_k = 2^k - 1$ 。

我們將以上關係畫成樹狀圖



由樹狀圖可得以下遞迴關係：

$$a_1 = 1$$

$$a_2 = a_1 + 1 = 2$$

$$a_3 = a_2 + 1 + (1 + 3 + 7) = 14$$

$$a_4 = a_3 + \left[\binom{1}{b_1 \text{項}} + \binom{1+3+7}{b_2 \text{項}} + \binom{1+3+7+\dots+(2^{15}-1)}{b_3 \text{項}} + \binom{1+3+7+\dots+(2^{65535}-1)}{b_4 \text{項}} + \dots \right. \\ \left. \dots + \binom{1+3+7+\dots+(2^{b_{14}}-1)}{b_{14} \text{項}} \right] = 14 + \left[\binom{2}{2-1} - 1 + \binom{3}{2-2} - 1 + \dots + \binom{15}{2-14} - 1 \right]$$

$$= 14 + \left(\binom{15}{2-14} - 14 \right) = \binom{15}{2} - 2$$

⋮

依此類推可得 $a_{k+1} = a_k + \underbrace{[S_1 + S_2 + \cdots + S_{a_k}]}_{\text{共 } b_k \text{ 項}}$ ，其中 $S_k = c_1 + c_2 + \cdots + c_{b_k}$ ，且 $c_k = 2^k - 1$ 、 $b_k = {}^k 2 - 1$ 。

問題 1：如何得知 $c_k = 2^k - 1$ 呢？

分析：觀察座標 $(\bigcirc, \square, c_k, 1)$ 變化為座標 $(\bigcirc', \square', c_{k+1}, 1)$ 的過程

$$(\bigcirc, \square, c_k, 1) \rightarrow (\bigcirc+1, \square+1, c_k+1, 1+2) \rightarrow \cdots \rightarrow (\bigcirc+h, \square+h, c_k+h, 1+2h) \xrightarrow{\text{迭代}} (\bigcirc+h, \square+h, c_{k+1}, 1)$$

其中 $c_k + h < 1 + 2h \Rightarrow h > c_k - 1$ 取 $h = c_k$ 得 $c_{k+1} = c_k + c_k + 1 = 2c_k + 1$
進位

即 $\langle c_k \rangle$ 滿足遞迴關係式 $\begin{cases} c_{k+1} = 2c_k + 1, k \geq 1 \\ c_1 = 1 \end{cases}$ ，解得 $c_k = 2^k - 1 (\forall k \in N)$ 。

問題 2：如何得知 $b_k = {}^k 2 - 1$ 呢？

分析：觀察座標 $(\bigcirc, b_k, 1, 1)$ 變化為 $(\bigcirc', b_{k+1}, 1, 1)$ 的過程

$$\begin{aligned} \text{由(3)的關係可知 } b_{k+1} &= b_k + (c_1 + c_2 + \cdots + c_{b_k}) + 1 = b_k + \underbrace{[(2^1 - 1) + (2^2 - 1) + \cdots + (2^{b_k} - 1)]}_{\text{進位}} + 1 \\ &= b_k + \frac{2(2^{b_k} - 1)}{2 - 1} - b_k + 1 = 2^{b_k+1} - 1 \end{aligned}$$

即 $\langle b_k \rangle$ 滿足遞迴關係式 $\begin{cases} b_{k+1} = 2^{b_k+1} - 1, k \geq 1 \\ b_1 = 1 \end{cases}$ ，解得 $b_k = {}^k 2 - 1 (\forall k \in N)$ 。

由以上可知

$$S_k = (2^1 - 1) + (2^2 - 1) + \cdots + (2^{b_k} - 1) = \frac{2(2^{b_k} - 1)}{2 - 1} - b_k = 2^{b_k+1} - 2 - b_k = 2^{k+1} - {}^k 2 - 1 = {}^{k+1} 2 - {}^k 2 - 1$$

$$\text{可推得 } a_{k+1} = a_k + \sum_{i=1}^{a_k} ({}^{i+1} 2 - {}^i 2 - 1) = a_k + ({}^{a_k+1} 2 - {}^1 2 - a_k) = {}^{a_k+1} 2 - 2$$

因此數列 $\langle a_k \rangle$ 的遞迴關係式為 $\begin{cases} a_{k+1} = {}^{a_k+1} 2 - 2, k \geq 1 \\ a_1 = 1 \end{cases}$

我們想要表示出一個很類似於定理 10 的公式，顯然，我們需要微調文獻[4]所定義的迭代對數。

定義 $\log^*(n)$ 滿足 $\log^*(n) = \begin{cases} 0, & n < 2 \\ 1 + \log^*(\log_2 n), & n \geq 2 \end{cases}$ 以及定義 $\log^{[4]}(n)$ 滿足

$$\log^{[4]}(n) = \begin{cases} 0, & n < 1 \\ 1 + \log^{[4]}(\log^*(n+2) - 1), & n \geq 1 \end{cases}, \text{ 則 } J_1^{[4]}(n) = \log^{[4]}(n)。$$

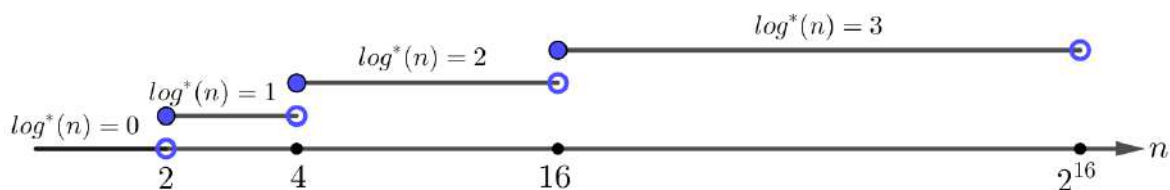
將結論整理如下：

4 維空間 $J_1^{[4]}(n)$ 之通式：

在 $\Delta_{n,1}^{[4]}$ 中，定義 $\log^{[4]}(n)$ 滿足 $\log^{[4]}(n) = \begin{cases} 0, & n < 1 \\ 1 + \log^{[4]}(\log^*(n+2) - 1), & n \geq 1 \end{cases}$ ，則

$$J_1^{[4]}(n) = \log^{[4]}(n)，\text{ 其中 } \log^*(n) \text{ 滿足 } \log^*(n) = \begin{cases} 0, & n < 2 \\ 1 + \log^*(\log_2 n), & n \geq 2 \end{cases}。$$

我們將 $\log^*(n)$ 的值整理於下圖，特別是 $\log^*(2^k) = k, \forall k \in \mathbb{N}$ 。



可以看得出來，無論是 $\log^*(n)$ 或 $\log^{[4]}(n)$ ，它們都是增加速度極為緩慢的函數。

(二) $m = 5, \ell = 1$

類似於「4 維空間斜列錯開法」，定義「5 維空間斜列錯開法」如下：

5 維空間斜列錯開法

在 $\Delta_{n,1}^{[5]}$ 中，設 5 維空間座標為 $(x_1, x_2, x_3, x_4, x_5)$ ， $x_1 = i_1$ 的紅球座標為 $(i_1, i_2, i_3, i_4, i_5)$ 。

令 $(i_1', i_2', i_3', i_4', i_5') = (i_1 + 1, i_2 + 1, i_3 + 1, i_4 + 1, i_5 + 2)$ ，進行以下迭代步驟，

- (1) 當 $i_1' < i_2'$ 時， $(i_1', i_2', i_3', i_4', i_5') \Rightarrow (i_1', i_2' - i_1', i_3', i_4', i_5')$ 再執行步驟(2)。否則直接步驟(2)。
- (2) 當 $i_2' < i_3'$ 時， $(i_1', i_2', i_3', i_4', i_5') \Rightarrow (i_1', i_2' + 1, i_3' - i_2', i_4', i_5')$ 再執行步驟(3)。否則直接步驟(3)。
- (3) 當 $i_3' < i_4'$ 時， $(i_1', i_2', i_3', i_4', i_5') \Rightarrow (i_1', i_2', i_3' + 1, i_4' - i_3', i_5')$ 再執行步驟(4)。否則直接步驟(4)。
- (4) 當 $i_4' < i_5'$ 時， $(i_1', i_2', i_3', i_4', i_5') \Rightarrow (i_1', i_2', i_3', i_4' + 1, i_5' - i_4')$ 回步驟(1)。

重複進行(1)(2)(3)(4)至 $i_1' \geq i_2' \geq i_3' \geq i_4' \geq i_5'$ 為止。此時 $(i_1', i_2', i_3', i_4', i_5')$ 為 $x_1 = i_1 + 1$ 的紅球坐標。

下表為以「5 維空間斜列錯開法」排列的 $\Delta_{n,1}^{[5]}$ 之紅球座標以及最佳陣列之最大值 k 。

k	1	2	3
紅	a_1	a_2	a_3
球	b_1	b_2	b_3
座	c_1	c_2	c_3
標	d_1	d_2	d_3
	e_1	e_2	e_3
	f_1	f_2	f_3
	g_1	g_2	g_3
	h_1	h_2	h_3
	i_1	i_2	i_3
	j_1	j_2	j_3
	k_1	k_2	k_3
	l_1	l_2	l_3
	m_1	m_2	m_3
	n_1	n_2	n_3
	o_1	o_2	o_3
	p_1	p_2	p_3
	q_1	q_2	q_3
	r_1	r_2	r_3
	s_1	s_2	s_3
	t_1	t_2	t_3
	u_1	u_2	u_3
	v_1	v_2	v_3
	w_1	w_2	w_3
	x_1	x_2	x_3
	y_1	y_2	y_3
	z_1	z_2	z_3
	aa_1	aa_2	aa_3
	ab_1	ab_2	ab_3
	ac_1	ac_2	ac_3
	ad_1	ad_2	ad_3
	ae_1	ae_2	ae_3
	af_1	af_2	af_3
	ag_1	ag_2	ag_3
	ah_1	ah_2	ah_3
	ai_1	ai_2	ai_3
	aj_1	aj_2	aj_3
	ak_1	ak_2	ak_3
	al_1	al_2	al_3
	am_1	am_2	am_3
	an_1	an_2	an_3
	ao_1	ao_2	ao_3
	ap_1	ap_2	ap_3
	aq_1	aq_2	aq_3
	ar_1	ar_2	ar_3
	as_1	as_2	as_3
	at_1	at_2	at_3
	au_1	au_2	au_3
	av_1	av_2	av_3
	aw_1	aw_2	aw_3
	ax_1	ax_2	ax_3
	ay_1	ay_2	ay_3
	az_1	az_2	az_3
	ba_1	ba_2	ba_3
	bb_1	bb_2	bb_3
	bc_1	bc_2	bc_3
	bd_1	bd_2	bd_3
	be_1	be_2	be_3
	bf_1	bf_2	bf_3
	bg_1	bg_2	bg_3
	bh_1	bh_2	bh_3
	bi_1	bi_2	bi_3
	bj_1	bj_2	bj_3
	bk_1	bk_2	bk_3
	bl_1	bl_2	bl_3
	bm_1	bm_2	bm_3
	bn_1	bn_2	bn_3
	bo_1	bo_2	bo_3
	bp_1	bp_2	bp_3
	bq_1	bq_2	bq_3
	br_1	br_2	br_3
	bs_1	bs_2	bs_3
	bt_1	bt_2	bt_3
	bu_1	bu_2	bu_3
	bv_1	bv_2	bv_3
	bw_1	bw_2	bw_3
	bx_1	bx_2	bx_3
	by_1	by_2	by_3
	bz_1	bz_2	bz_3
	ca_1	ca_2	ca_3
	cb_1	cb_2	cb_3
	cc_1	cc_2	cc_3
	cd_1	cd_2	cd_3
	ce_1	ce_2	ce_3
	cf_1	cf_2	cf_3
	cg_1	cg_2	cg_3
	ch_1	ch_2	ch_3
	ci_1	ci_2	ci_3
	cj_1	cj_2	cj_3
	ck_1	ck_2	ck_3
	cl_1	cl_2	cl_3
	cm_1	cm_2	cm_3
	cn_1	cn_2	cn_3
	co_1	co_2	co_3
	cp_1	cp_2	cp_3
	cq_1	cq_2	cq_3
	cr_1	cr_2	cr_3
	cs_1	cs_2	cs_3
	ct_1	ct_2	ct_3
	cu_1	cu_2	cu_3
	cv_1	cv_2	cv_3
	cw_1	cw_2	cw_3
	cx_1	cx_2	cx_3
	cy_1	cy_2	cy_3
	cz_1	cz_2	cz_3
	da_1	da_2	da_3
	db_1	db_2	db_3
	dc_1	dc_2	dc_3
	dd_1	dd_2	dd_3
	de_1	de_2	de_3
	df_1	df_2	df_3
	dg_1	dg_2	dg_3
	dh_1	dh_2	dh_3
	di_1	di_2	di_3
	dj_1	dj_2	dj_3
	dk_1	dk_2	dk_3
	dl_1	dl_2	dl_3
	dm_1	dm_2	dm_3
	dn_1	dn_2	dn_3
	do_1	do_2	do_3
	dp_1	dp_2	dp_3
	dq_1	dq_2	dq_3
	dr_1	dr_2	dr_3
	ds_1	ds_2	ds_3
	dt_1	dt_2	dt_3
	du_1	du_2	du_3
	dv_1	dv_2	dv_3
	dw_1	dw_2	dw_3
	dx_1	dx_2	dx_3
	dy_1	dy_2	dy_3
	dz_1	dz_2	dz_3
	ea_1	ea_2	ea_3
	eb_1	eb_2	eb_3
	ec_1	ec_2	ec_3
	ed_1	ed_2	ed_3
	ee_1	ee_2	ee_3
	ef_1	ef_2	ef_3
	eg_1	eg_2	eg_3
	eh_1	eh_2	eh_3
	ei_1	ei_2	ei_3
	ej_1	ej_2	ej_3
	ek_1	ek_2	ek_3
	el_1	el_2	el_3
	em_1	em_2	em_3
	en_1	en_2	en_3
	eo_1	eo_2	eo_3
	ep_1	ep_2	ep_3
	eq_1	eq_2	eq_3
	er_1	er_2	er_3
	es_1	es_2	es_3
	et_1	et_2	et_3
	eu_1	eu_2	eu_3
	ev_1	ev_2	ev_3
	ew_1	ew_2	ew_3
	ex_1	ex_2	ex_3
	ey_1	ey_2	ey_3
	ez_1	ez_2	ez_3
	fa_1	fa_2	fa_3
	fb_1	fb_2	fb_3
	fc_1	fc_2	fc_3
	fd_1	fd_2	fd_3
	fe_1	fe_2	fe_3
	ff_1	ff_2	ff_3
	fg_1	fg_2	fg_3
	fh_1	fh_2	fh_3
	fi_1	fi_2	fi_3
	fj_1	fj_2	fj_3
	fk_1	fk_2	fk_3
	fl_1	fl_2	fl_3
	fm_1	fm_2	fm_3
	fn_1	fn_2	fn_3
	fo_1	fo_2	fo_3
	fp_1	fp_2	fp_3
	fq_1	fq_2	fq_3
	fr_1	fr_2	fr_3
	fs_1	fs_2	fs_3
	ft_1	ft_2	ft_3
	fu_1	fu_2	fu_3
	fv_1	fv_2	fv_3
	fw_1	fw_2	fw_3
	fx_1	fx_2	fx_3
	fy_1	fy_2	fy_3
	fz_1	fz_2	fz_3
	ga_1	ga_2	ga_3
	gb_1	gb_2	gb_3
	gc_1	gc_2	gc_3
	gd_1	gd_2	gd_3
	ge_1	ge_2	ge_3
	gf_1	gf_2	gf_3
	gg_1	gg_2	gg_3
	gh_1	gh_2	gh_3
	gi_1	gi_2	gi_3
	gj_1	gj_2	gj_3
	gk_1	gk_2	gk_3
	gl_1	gl_2	gl_3
	gm_1	gm_2	gm_3
	gn_1	gn_2	gn_3
	go_1	go_2	go_3
	gp_1	gp_2	gp_3
	gq_1	gq_2	gq_3
	gr_1	gr_2	gr_3
	gs_1	gs_2	gs_3
	gt_1	gt_2	gt_3
	gu_1	gu_2	gu_3
	gv_1	gv_2	gv_3
	gw_1	gw_2	gw_3
	gx_1	gx_2	gx_3
	gy_1	gy_2	gy_3
	gz_1	gz_2	gz_3
	ha_1	ha_2	ha_3
	hb_1	hb_2	hb_3
	hc_1	hc_2	hc_3
	hd_1	hd_2	hd_3
	he_1	he_2	he_3
	hf_1	hf_2	hf_3
	hg_1	hg_2	hg_3
	hh_1	hh_2	hh_3
	hi_1	hi_2	hi_3
	hj_1	hj_2	hj_3
	hk_1	hk_2	hk_3
	hl_1	hl_2	hl_3
	hm_1	hm_2	hm_3
	hn_1	hn_2	hn_3
	ho_1	ho_2	ho_3
	hp_1	hp_2	hp_3
	hq_1	hq_2	hq_3
	hr_1	hr_2	hr_3
	hs_1	hs_2	hs_3
	ht_1	ht_2	ht_3
	hu_1	hu_2	hu_3
	hv_1	hv_2	hv_3
	hw_1	hw_2	hw_3
	hx_1	hx_2	hx_3
	hy_1	hy_2	hy_3
	hz_1	hz_2	hz_3
	ia_1	ia_2	ia_3
	ib_1	ib_2	ib_3
	ic_1	ic_2	ic_3
	id_1	id_2	id_3

與 4 維的情形相同，每個紅球座標間都存在著某種意義與關係。敘述如下：

(1) 數列 $\langle a_k \rangle$ 滿足 $(a_1, 1, 1, 1, 1) \rightarrow (a_2, 1, 1, 1, 1) \rightarrow (a_3, 1, 1, 1, 1) \rightarrow \dots$

(2) 數列 $\langle b_k \rangle$ 滿足 $(a_i, b_1, 1, 1, 1) \rightarrow (O', b_2, 1, 1, 1) \rightarrow (O'', b_3, 1, 1, 1) \rightarrow \dots$ 共 a_i 項

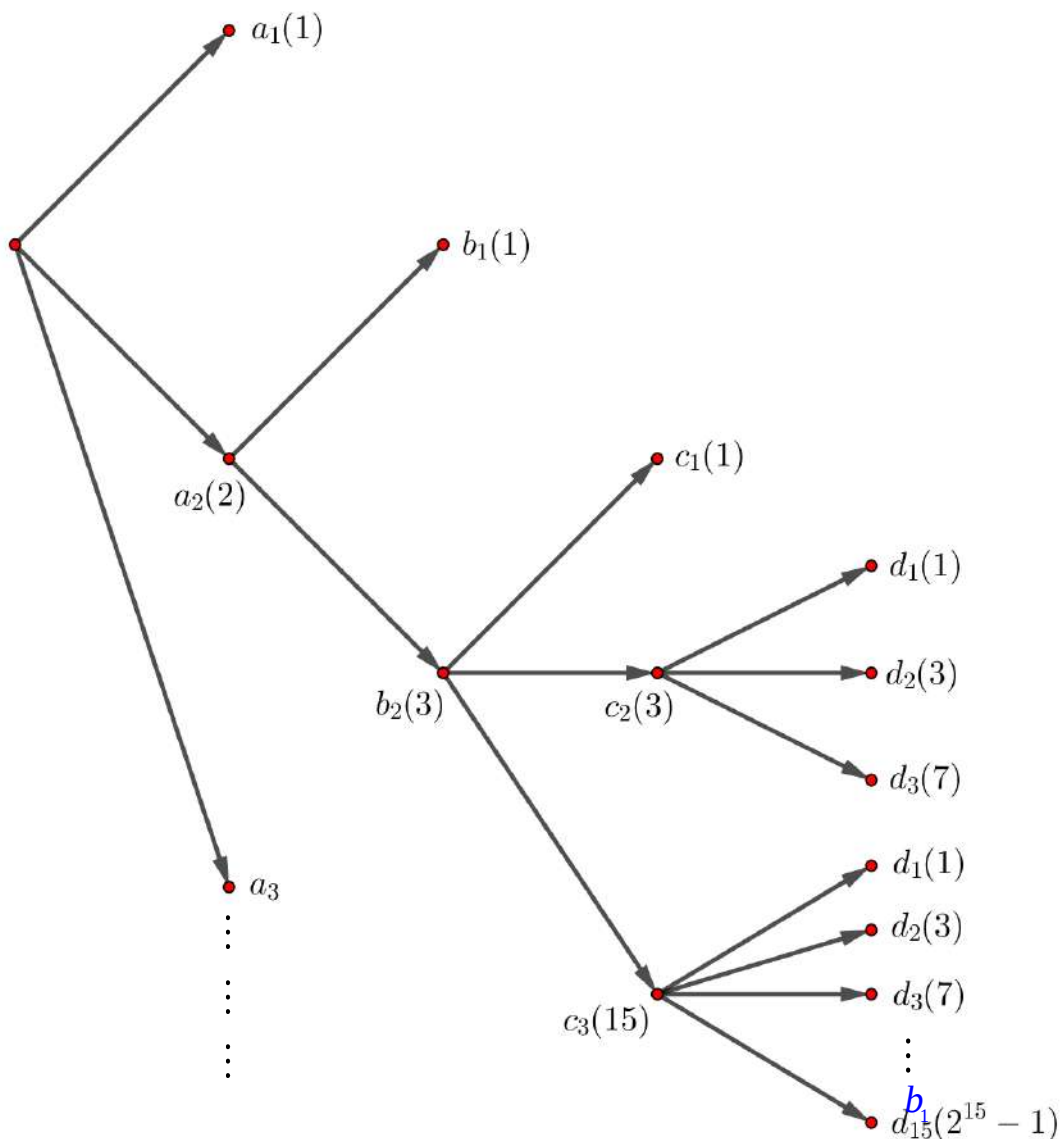
(3) 數列 $\langle c_k \rangle$ 滿足 $(O, b_i, c_1, 1, 1) \rightarrow (O', \square, c_2, 1, 1) \rightarrow (O'', \square', c_3, 1, 1) \rightarrow \dots$ 共 b_i 項

(4) 數列 $\langle d_k \rangle$ 滿足 $(O, \square, c_i, d_1, 1) \rightarrow (O', \square', \Delta, d_2, 1) \rightarrow (O'', \square'', \Delta', d_3, 1) \rightarrow \dots$ 共 c_i 項

而這些 d_k 值代表其所領導的小區域之紅球數量，即若紅球座標變化為

$(O, \square, \Delta, d_k, 1) \rightarrow (O', \square', \Delta', d_{k+1}, 1)$ ，則 $O' = O + d_k$ 。

我們將以上關係畫成樹狀圖



由樹狀圖可得以下遞迴關係：

$$a_1 = 1$$

$$a_2 = a_1 + 1 = 2$$

$$a_3 = a_2 + \underbrace{\left[\underbrace{\binom{1}{b_1 \text{項}}} + \underbrace{\left(\underbrace{\binom{1}{c_1 \text{項}}} + \underbrace{(1+3+7)}_{c_2 \text{項}} + \underbrace{(1+3+7+\cdots+(2^{c_3}-1))}_{c_3 \text{項}} \right)}_{b_2 \text{項}} \right]}_{a_2 \text{項}}$$

$$= 2 + (1) + ((^2 2 - ^1 2 - 1) + (^3 2 - ^2 2 - 1) + (^4 2 - ^3 2 - 1)) = 2^{16} - 2 = 65534$$

$$\vdots$$

依此類推可得 $a_{k+1} = a_k + [S_1 + S_2 + \cdots + S_{a_k}]$ ，其中 $S_k = \underbrace{T_1 + T_2 + \cdots + T_{b_k}}_{\text{共 } b_k \text{ 項}}$ 、 $T_k = \underbrace{d_1 + d_2 + \cdots + d_{c_k}}_{\text{共 } c_k \text{ 項}}$ ，

且 $d_k = 2^k - 1$ 、 $c_k = {}^k 2 - 1$ 、 $b_k = {}_k 2 - 1$ 。

【註】 ${}_b a$ 為五級運算，是迭代幕次的迭代，定義為 ${}_b a = \overset{a}{\underset{b}{\dots}} a$ ，亦可以符號 $a[5]b$ 表示。

同理四維情形問題 1、2 之分析，我們亦可推得 $d_k = 2^k - 1$ 、 $c_k = {}^k 2 - 1$ 。接著分析 b_k 的通式。

問題 3：如何得知 $b_k = {}_k 2 - 1$ 呢？

分析：觀察座標 $(O, b_k, 1, 1, 1)$ 變化為 $(O', b_{k+1}, 1, 1, 1)$ 的過程

$$\text{已知 } T_k = \sum_{i=1}^{c_k} d_i = \sum_{i=1}^{c_k} (2^i - 1) = \frac{2(2^{c_k} - 1)}{2 - 1} - c_k = 2^{c_k+1} - 2 - c_k = 2^{^k 2} - {}^k 2 - 1 = {}^{k+1} 2 - {}^k 2 - 1$$

$$b_{k+1} = b_k + \underbrace{(T_1 + T_2 + \cdots + T_{b_k})}_{\text{進位}} + 1 = b_k + [(^2 2 - ^1 2 - 1) + (^3 2 - ^2 2 - 1) + \cdots + ({}^{b_k+1} 2 - {}^{b_k} 2 - 1)] + 1$$

$$= b_k + {}^{b_k+1} 2 - ^1 2 - b_k + 1 = {}^{b_k+1} 2 - 1$$

$$\text{即 } \langle b_k \rangle \text{ 滿足遞迴關係式 } \begin{cases} b_{k+1} = {}^{b_k+1} 2 - 1, k \geq 1 \\ b_1 = 1 \end{cases}, \text{ 解得 } b_k = {}_k 2 - 1 = 2[5]k - 1 (\forall n \in N)。$$

$$\text{由以上可知 } S_k = \sum_{i=1}^{b_k} ({}^{i+1} 2 - {}^i 2 - 1) = {}^{b_k+1} 2 - 2 - b_k = {}^{^k 2} 2 - {}^k 2 - 1 = {}^{k+1} 2 - {}^k 2 - 1$$

$$\text{可推得 } a_{k+1} = a_k + \sum_{i=1}^{a_k} ({}^{i+1} 2 - {}^i 2 - 1) = a_k + {}^{a_k+1} 2 - {}_1 2 - a_k = {}^{a_k+1} 2 - 2$$

$$\text{因此數列 } \langle a_k \rangle \text{ 的遞迴關係式為 } \begin{cases} a_{k+1} = {}^{a_k+1} 2 - 2, k \geq 1 \\ a_1 = 1 \end{cases}$$

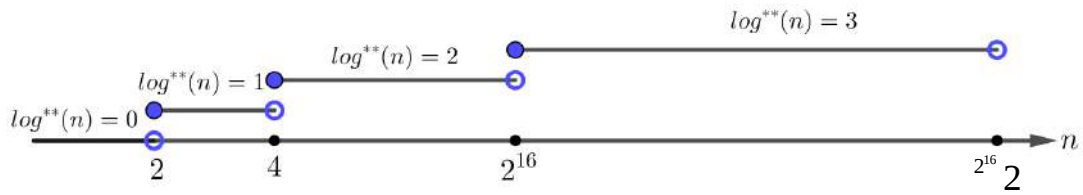
由以上結論，我們可以得到 $J_1^{[5]}(n)$ 之通式，敘述如下：

5 維空間 $J_1^{[5]}(n)$ 之通式：

$$\text{在 } \Delta_{n,1}^{[5]} \text{ 中，定義 } \log^{[5]}(n) \text{ 滿足 } \log^{[5]}(n) = \begin{cases} 0, & n < 1 \\ 1 + \log^{[5]}(\log^{**}(n+2) - 1), & n \geq 1 \end{cases}, \text{ 則}$$

$$J_1^{[5]}(n) = \log^{[5]}(n)。 \text{ 其中 } \log^{**}(n) \text{ 滿足 } \log^{**}(n) = \begin{cases} 0, & n < 2 \\ 1 + \log^{**}(\log^*(n)), & n \geq 2 \end{cases}。$$

我們將 $\log^{**}(n)$ 的值整理如下，特別的是 $\log^{**}(2) = k, \forall k \in N$ 。



$$\text{可得 } \log^{[5]}(n) = \begin{cases} 1, & n=1 \\ 2, & 2 \leq n < 2^{16} - 2 \\ 3, & 2^{16} - 2 \leq n < {}_{65535}2 - 2 \\ \vdots \end{cases}。$$

可以看得出來，無論是 $\log^{**}(n)$ 或 $\log^{[5]}(n)$ ，它們都是增加速度極為緩慢的函數。

(三) $m \in N$ 、 $\ell=1$

事實上，4 維與 5 維空間通式的推導過程也可以類推至 m 維情形，我們將結論整理如下：

m 維空間斜列錯開法

在 $\Delta_{n,1}^{[m]}$ 中，設 m 維空間座標為 $(x_1, x_2, \dots, x_m)$ ， $x_1 = i_1$ 的紅球座標為 $(i_1, i_2, \dots, i_m)$ 。

令 $(i_1', i_2', \dots, i_{m-1}', i_m') = (i_1 + 1, i_2 + 1, \dots, i_{m-1} + 1, i_m + 2)$ ，進行以下迭代步驟，

(1) 當 $i_1' < i_2'$ 時， $(i_1', i_2', \dots, i_m') \Rightarrow (i_1', i_2' - i_1', \dots, i_m')$ 再執行步驟(2)。否則直接步驟(2)。

(2) 當 $i_2' < i_3'$ 時， $(i_1', i_2', i_3', \dots, i_m') \Rightarrow (i_1', i_2' + 1, i_3' - i_2', \dots, i_m')$ 再執行步驟(3)。否則直接步驟(3)。

⋮

($m-1$) 當 $i_{m-1}' < i_m'$ 時， $(i_1', i_2', \dots, i_{m-1}', i_m') \Rightarrow (i_1', i_2', \dots, i_{m-1}' + 1, i_m' - i_{m-1}')$ 回步驟(1)。

重複進行(1)(2)⋯($m-1$)至 $i_1' \geq i_2' \geq \dots \geq i_{m-1}' \geq i_m'$ 為止。此時 $(i_1', i_2', \dots, i_{m-1}', i_m')$ 為 $x_1 = i_1 + 1$ 的紅球坐標。

每個紅球座標間都存在著某種意義與關係。

(1) 數列 $\langle a_{1,k} \rangle$ 滿足 $(a_{1,1}, \underbrace{1, 1, \dots, 1}_{m-1}) \rightarrow (a_{1,2}, \underbrace{1, 1, \dots, 1}_{m-1}) \rightarrow (a_{1,3}, \underbrace{1, 1, \dots, 1}_{m-1}) \rightarrow \dots$

(2) 數列 $\langle a_{2,k} \rangle$ 滿足 $(a_{1,i}, \underbrace{a_{2,1}, 1, \dots, 1}_{m-2}) \rightarrow (\bigcirc', \underbrace{a_{2,2}, 1, \dots, 1}_{m-2}) \rightarrow (\bigcirc'', \underbrace{a_{2,3}, 1, \dots, 1}_{m-2}) \rightarrow \dots$ 共 $a_{1,i}$ 項

(3) 數列 $\langle a_{3,k} \rangle$ 滿足 $(\bigcirc, \underbrace{a_{2,i}, a_{3,1}, 1, \dots, 1}_{m-3}) \rightarrow (\bigcirc', \underbrace{\square, a_{3,2}, 1, \dots, 1}_{m-3}) \rightarrow (\bigcirc'', \underbrace{\square', a_{3,3}, 1, \dots, 1}_{m-3}) \rightarrow \dots$ 共 $a_{2,i}$ 項

⋮

($m-1$) 數列 $\langle a_{m-1,k} \rangle$ 滿足 $(\bigcirc, \underbrace{\square, \dots, a_{m-2,i}, a_{m-1,1}}_{m-2}) \rightarrow (\bigcirc', \underbrace{\square', \dots, \Delta, a_{m-1,2}}_{m-2}) \rightarrow (\bigcirc'', \underbrace{\square'', \dots, \Delta', a_{m-1,3}}_{m-2}) \rightarrow \dots$

共 $a_{m-2,i}$ 項，而這些 $a_{m-1,k}$ 值代表其所領導的小區域之紅球數量，即若紅球座標變化為

$(\bigcirc, \square, \dots, \Delta, a_{m-1,k}, 1) \rightarrow (\bigcirc', \square', \dots, \Delta', a_{m-1,k+1}, 1)$ ，則 $\bigcirc' = \bigcirc + a_{m-1,k}$ 。

由以上迭代關係可知 $a_{m-1,k} = 2^k - 1 = 2[3]k - 1$ 、 $a_{m-2,k} = {}^k 2 - 1 = 2[4]k - 1$ 、 $a_{m-3,k} = {}^k 2 - 1 = 2[5]k - 1$ 、……，可歸納出通式 $a_{i,k} = 2[m-i+2]k - 1$ 。

【註】 $a[t]b$ 代表第 t 級(超)運算，每一級的超運算都是對上一級的迭代。即

$$a[t]b = \underbrace{a[t-1](a[t-1](a[t-1] \cdots (a[t-1](a[t-1]a))) \cdots)}_b$$

因此，我們可以歸納出 m 維空間 $J_1^{[m]}(n)$ 之通式如下：

m 維空間 $J_1^{[m]}(n)$ 之通式：

$$\text{在 } \Delta_{n,1}^{[m]} (m \geq 4) \text{ 中，定義 } \log^{[m]}(n) \text{ 滿足 } \log^{[m]}(n) = \begin{cases} 0 & , n < 1 \\ 1 + \log^{[m]} \left(\log^{* \cdots * (n+2)} - 1 \right) & , n \geq 1 \end{cases} \text{，則}$$

$$J_1^{[m]}(n) = \log^{[m]}(n) \circ \text{其中 } \log^{* \cdots * (n)} \text{ 滿足 } \log^{* \cdots * (n)} = \begin{cases} 0 & , n < 2 \\ 1 + \log^{* \cdots *} \left(\log^{* \cdots * (n)} \right) & , n \geq 2 \end{cases} \circ$$

事實上，二維與三維的情形也符合此通式，若 $\log_2(n)$ 視作為 $\log^*(n)$ 的低一階運算，且 $\frac{n}{2}$ 視作為 $\log_2(n)$ 的低一階運算。

$$\text{在三維的情形，若我們定義 } \log^{[3]}(n) \text{ 滿足 } \log^{[3]}(n) = \begin{cases} 0 & , n < 1 \\ 1 + \log^{[3]}(\log_2(n+2) - 1) & , n \geq 1 \end{cases} \text{，則}$$

$J_1^{[3]}(n) = \log^{[3]}(n)$ 。這個結論和定理 10 是相符的。

$$\text{在二維的情形，若我們定義 } \log^{[2]}(n) \text{ 滿足 } \log^{[2]}(n) = \begin{cases} 0 & , n < 1 \\ 1 + \log^{[2]} \left(\frac{n+2}{2} - 1 \right) & , n \geq 1 \end{cases} \text{，則}$$

$J_1^{[2]}(n) = \log^{[2]}(n)$ 。這個結論和定理 2 也是相符的。我們成功將 $\ell = 1$ 的情形推廣至 m 維空間。

參、研究結果

一、 $\Delta_{n,\ell}$ 之最佳路徑最小化問題

在 $\Delta_{n,\ell}$ 中，可以「平面斜列錯開法」建構出任意大小的最糟日式三角形，令數列 $\langle a_k \rangle$ 滿足

$$\begin{cases} a_k = 1 + \left\lfloor \frac{\sum_{i=1}^{k-1} a_i + \frac{(\ell-1)(\ell-2)}{2}}{\ell} \right\rfloor, k \geq \ell+1 \\ a_1 = 1, a_2 = 2, \dots, a_\ell = \ell \end{cases} \text{，若 } a_k \leq n < a_{k+1} \text{，則 } J_\ell(n) = k \text{。特別的是，當 } \ell = 1 \text{ 時，}$$

$$J_1(n) = \lceil \log_2 n \rceil + 1, \forall n \in N \circ$$

二、 $\Delta_{n,\ell}$ 之最糟路徑最大化問題

在 $\Delta_{n,\ell}$ 中，可以「平面斜列防堵法」建構出任意大小的最佳日式三角形，當 $\ell=1$ 時，

$$I_1(n)=1, \forall n \in N, \text{ 當 } \ell \geq 2 \text{ 時，令數列 } \langle a_k \rangle \text{ 滿足 } \begin{cases} a_k = 1 + \left\lfloor \frac{\sum_{i=1}^{k-1} a_i + \frac{(\ell-1)(\ell-2)}{2}}{\ell-1} \right\rfloor, k \geq \ell+1 \\ a_1 = 1, a_2 = 2, \dots, a_\ell = \ell \end{cases}, \text{ 若}$$

$a_k \leq n < a_{k+1}$ ，則 $I_\ell(n) = k$ 。特別的是，當 $\ell=2$ 時， $I_2(n) = \lceil \log_2 n \rceil + 1, \forall n \in N$ 。

三、 $\Delta_{n,\ell}^*$ 之最佳路徑最小化問題

在 $\Delta_{n,\ell}^*$ 中，可以「空間斜列錯開法」建構出任意大小的最糟日式三角垛，且可以迭代演算

$$\text{法求出 } J_\ell^*(n) \text{。特別的是，當 } \ell=1 \text{ 時，定義 } \log^{[3]}(n) \text{ 滿足 } \log^{[3]}(n) = \begin{cases} 0, & n < 1 \\ 1 + \log^{[3]}(\log_2(n+2)-1), & n \geq 1 \end{cases}$$

，則 $J_1^*(n) = \log^{[3]}(n)$ 。

四、 $\Delta_{n,\ell}^*$ 之最糟路徑最大化問題

在 $\Delta_{n,\ell}^*$ 中，可以「空間斜列防堵法」建構出任意大小的最佳日式三角垛，且可以迭代演算

$$\text{法求出 } I_\ell^*(n) \text{ 特別的是，當 } \ell=1, 2 \text{ 時， } I_1^*(n) = I_2^*(n) = 1, \text{ 當 } \ell=3 \text{ 時， } I_3^*(n) = \begin{cases} 1, & n=1 \\ 2, & n \geq 2 \end{cases} \text{。}$$

更進一步，當 n 足夠大時， $I_\ell^*(n)$ 會呈現定值，即存在 $i, c \in N$ ，使得 $I_\ell^*(n) = c, \forall n \geq i$ ，而 i

的最小值 $\min_\ell(i)$ 滿足 $\ell-1 \leq \min_\ell(i) \leq \ell-2 + \sum_{k=1}^{\ell-2} \left\lceil \frac{\ell-2}{k} \right\rceil$ 。

五、 $\ell=1$ 高維空間情形之探討

$$\text{在 } \Delta_{n,1}^{[m]} (m \geq 4) \text{ 中，定義 } \log^{[m]}(n) \text{ 滿足 } \log^{[m]}(n) = \begin{cases} 0, & n < 1 \\ 1 + \log^{[m]} \left(\log^{* \dots * (n+2)} - 1 \right), & n \geq 1 \end{cases}, \text{ 則}$$

$$J_1^{[m]}(n) = \log^{[m]}(n) \text{。其中 } \log^{* \dots * (n)} \text{ 滿足 } \log^{* \dots * (n)} = \begin{cases} 0, & n < 2 \\ 1 + \log^{* \dots *} \left(\log^{* \dots * (n)} \right), & n \geq 2 \end{cases} \text{。}$$

肆、討論與未來展望

在這篇研究中，我們除了針對平面日式三角形作延伸探討，推導出 $J_\ell(n)$ 與 $I_\ell(n)$ 的通式，同時配合迭代演算法，將問題推廣至空間日式三角垛的情形，大大的優化了程式的時間複雜度。其中，我們對於 $\ell=1$ 的情形特別感到興趣，比如在 $\Delta_{n,1}$ 中， $J_1(n) = \lceil \log_2 n \rceil + 1$ 代表著 n 可以執行幾次「加 2 後除以 2 再減 1」的運算後小於 1。而在 $\Delta_{n,1}^*$ 中， $J_1^*(n) = \log^{[3]}(n)$ 代表著 n 可以執行幾次「加 2 後取以 2 為底的對數值再減 1」的運算後小於 1。最終我們成功地推導出高維的通式， $J_1^{[m]}(n)$ 值代表 n 可以執行幾次「加 2 後取 $\log^{* \dots *}$ 值再減 1」的運算後小於 1，它是一個增加速度極為緩慢的函數。除此之外，在 $\Delta_{n,\ell}^*$ 之最糟路徑最大化問題中，我們已找到 $I_\ell^*(n)$ 開始呈現定值的時機，同時，針對 $I_\ell^*(n)$ 開始呈現定值的最小 n 值，找到它的一組上界與下界，是否能夠再估計出更精確的上下界，是未來努力的目標。另一方面，最糟日式三角形(垛)與最佳日式三角形(垛)的個數，也是個值得研究的排列組合問題。

伍、參考文獻資料

1. 64th International Mathematical Olympiad. 取自 <https://www.imo-official.org/problems.aspx>。
2. Eric Shen(2023). IMO2023. 取自 <http://ericshen.net/exams/IMO2023.pdf>。
3. A005428. 取自 The On-Line Encyclopedia of Integer equences®。
4. Iterated logarithm.。取自 https://en.wikipedia.org/wiki/Iterated_logarithm。
5. Hyperoperation. 取自 <https://en.wikipedia.org/wiki/Hyperoperation>。

附錄

【程式碼 — $J^*(n)$ 】

```
#include<bits/stdc++.h>
using namespace std;
#pragma GCC target("avx2,popcnt")
#pragma GCC optimize("O3,unroll-loops,fast-math")
#define IOS() ios::sync_with_stdio(0),cin.tie(0)
void solve(int n,int l) {
    vector<int> a;
    int x=1,y=1,z=1,k=0;//x:i_t y:j_t z:k_t
    while(1) {
        if(((long long)y*(y-1)>>1)+z<=1) {
            k++;
            a.push_back(x);
        }
        if(x>=n) break;
        if(((long long)(x+1)*(x+2)>>1)<1) {
            x=y=z=x+1;
            continue;
        }
        x++,y++,z+=l+1;
        while(1) {
            if(x<y) y-=x;
            else if(y<z) z-=y,y++;
            else break;
        }
    }
    for(int i=0;i<a.size();i++) cout<<a[i]<<' ';cout<<'\\n';
    cout<<"J("<<n<<","<<l<<") = "<<k<<endl;
}
signed main() {
    IOS();
    int n,l;
    while(cin>>n>>l) {
        solve(n,l);
    }
    return 0;
}
```

【程式碼 — $I^*(n)$ 】

```
#include<bits/stdc++.h>
using namespace std;
#pragma GCC target("avx2,popcnt")
#pragma GCC optimize("O3,unroll-loops,fast-math")
#define IOS() ios::sync_with_stdio(0),cin.tie(0)
void solve(int n,int l) {
    vector<int> a;
    int x=1,y=1,z=1,k=0;//x:i_t y:j_t z:k_t
    while(1) {
        if(((long long)y*(y-1)>>1)+z<=1) {
            k++;
            a.push_back(x);
        }
        if(x>=n) break;
        if(((long long)(x+1)*(x+2)>>1)<1) {
            x=y=z=x+1;
            continue;
        }
        z-=1;
        while(1) {
            if(y<1) y+=x,z+=x;
            else if(z<1) z+=y-1,y--;
            else break;
        }
        x++,y++,z++;
    }
    for(int i=0;i<a.size();i++) cout<<a[i]<<' ';cout<<'\\n';
    cout<<"I("<<n<<","<<l<<") = "<<k<<endl;
}
signed main() {
    IOS();
    int n,l;
    while(cin>>n>>l) {
        solve(n,l);
    }
    return 0;
}
```

【評語】 010035

將一個 IMO 題目做延伸，作者用其方法給出問題解，也將題目延伸至三角垛及高維 $l=1$ 的情形，予以解決，對於高維情形相對複雜，可先用電腦程式試著找出是否有一規律性，然後再予以解決，除此之外，將來可在排列個數的變化上再予以著墨。